รถติดตามส่วนบุคคล

CarTracker

นายเตชะเวช ภักดีณรงค์

นายธนโชติ คำเพ็ชรดี

นายธัญวุฒิ หิรัญประสาทกุล

นายปิติพล เอมอินทร์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถติดตามส่วนบุคคล

CarTracker

นายเตชะเวช ภักดีณรงค์

นายธนโชติ คำเพ็ชรดี

นายธัญวุฒิ หิรัญประสาทกุล

นายปิติพล เอมอินทร์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CarTracker

Techawet Pagdinaronk
Thanachot kampechdee
Tanywut hiranprasatkun
Pitipol Aemin

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2021

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

หัวข้อปริญญานิพนธ์ รถติดตามส่วนบุคคล

Thesis Title CarTracker

นักศึกษาผู้จัดทำ	นายเตชะเวช ภักดีณรงค์	**รหัสนักศึกษา**	62015042
	นายธนโชติ คำเพ็ชรดี	**รหัสนักศึกษา**	62015047
	นายธัญวุฒิ หิรัญประสาทกุล	**รหัสนักศึกษา**	62015057
	นายปิติพล เอมอินทร์	**รหัสนักศึกษา**	62015075

ปริญญา วิศวกรรมศาสตรบัณฑิต

สาขาวิชา วิศวกรรมการวัดคุม

ปีการศึกษา 2564

อาจารย์ที่ปรึกษาปริญญานิพนธ์	**ลายมือชื่อ**
ผศ.ดร.นภศูล วงษ์วานิช	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	รถติดตามส่วนบุคคล			
นักศึกษาผู้จัดทำ	นายเตชะเวช	ภักดีณรงค์	รหัสนักศึกษา	62015042
	นายธนโชติ	คำเพ็ชรดี	รหัสนักศึกษา	62015047
	นายธัญวุฒิ	หิรัญประสาทกุล	รหัสนักศึกษา	62015057
	นายปิติพล	เอมอินทร์	รหัสนักศึกษา	62015075
ปริญญา	วิศวกรรมศาสตรบัณฑิต			
สาขาวิชา	วิศวกรรมการวัดคุม			
ปีการศึกษา	2564			
อาจารย์ที่ปรึกษา	ผศ.ดร.นภศูล วงษ์วานิช			

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการสร้าง CarTracker ซึ่งจำเป็นต้องศึกษาตั้งแต่ส่วนประกอบของ arduino อุปกรณ์ และการนำมาใช้งาน การใช้งานการเขียนโปรแกรม รวมทั้งการนำความรู้ที่ได้จากการเรียนที่ผ่านมามาประยุกต์ใช้ให้เกิดเป็นโครงงานดังกล่าว สามารถควบคุมผ่านโทรศัพท์มือถือโดยใช้แอพพลิเคชั่นควบคุม โดยเชื่อมต่อกับ Bluetooth กำหนด CarTracker นำทิศทางโดยใช้ GPS เข้ามาประกอบช่วยกับการทำงานของ CarTracker ให้สามารถติดตามคนใช้งานและแอพพลิเคชันควบคุม วัตถุประสงค์เพื่อใช้หลักการทำงานแบบไร้คนบังคับ เป็นแบบไร้สายผ่านแอพในการควบคุม เพื่อเป็นสื่อกลางในการเรียนการสอนในวิชาที่ต้องใช้ arduino เป็นหลัก เพื่อเป็นแนวทางในการสร้างสิ่งประดิษฐ์พื้นฐานในการใช้อุปกรณ์และโปรแกรมในการสร้าง CarTracker

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	CarTracker		
Student	Mr. Techawet	Pagdinaronk	Student ID. 62015042
	Mr. Thanachot	Kampechdee	Student ID. 62015047
	Mr. Tanyawut	Hiranprasatkun	Student ID. 62015057
	Mr. Pitipol	Aemin	Student ID. 62015075
Degree	Bachelor of Engineering		
Program	Instrumentation Engineering		
Year	2021		
Thesis Advisor	Asst. Prof. Dr. Nopasool Wongvanich		

ABSTRACT

This project is the creation of CarTracker, which is necessary to study from arduino components, equipment and usage. programming usage including applying the knowledge gained from past studies to create such projects Can be controlled via mobile phone using the control application. By connecting to Bluetooth, set the CarTracker directions using GPS to help with the work of CarTracker to be able to track users and control applications. The objective is to use the unmanned working principle. It is wireless via app to control. To be a medium for teaching in subjects that use arduino as a guideline for creating basic inventions in using equipment and programs to create CarTracke.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี เพราะได้รับคำปรึกษาและความอนุเคราะห์จากอาจารย์ที่ปรึกษา ผศ.ดร.นภศูล วงษ์วานิช อีกทั้งยังเอื้อเฟื้ออุปกรณ์และเครื่องมือต่าง ๆ ในการทำปริญญานิพนธ์นี้ ผู้จัดทำขอขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ อาจารย์สาขาวิชาวิศวกรรมการวัดและควบคุมทุกท่าน ที่ให้คำแนะนำและถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า อันเป็นประโยชน์ต่อการทำปริญญานิพนธ์ฉบับนี้

และที่ลืมเสียไม่ได้คือ ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ซึ่งเป็นที่รักและเคารพยิ่ง ที่สนับสนุนและเป็นแรงบรรดาลใจในการทำปริญญานิพนธ์ฉบับนี้ คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ผู้วิจัยขอมอบแด่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญรูป VI

บทที่ 1 บทนำ 1
1.1 ความเป็นมาและความสำคัญของปัญหา 1
1.2 วัตถุประสงค์ของปริญญานิพนธ์ 1
1.3 ขอบเขตของปริญญานิพนธ์ 2
1.4 ประโยชน์ที่คาดว่าจะได้รับ 2
1.5 ขั้นตอนการศึกษา 2

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 3
2.1 บอร์ดไมโครคอนโทรลเลอร์ 3
2.1.1.1 ใช้แรงดันไฟฟ้าเลี้ยงไอซีไมโครคอนโทรลเลอร์ 3
2.1.1.2 แรงดันไฟฟ้าเลี้ยงไอซีไมโครคอนโทรลเลอร์ 3
2.1.1.3 แรงดันไฟฟ้าป้อนที่บอร์ดไมโครคอนโทรลเลอร์ 3
2.1.1.4 มีพอร์ตดิจิตอลอินพุต/เอาต์พุต 3
2.1.1.5 มีพอร์ตอนาล็อกอินพุต 3
2.1.1.6 สามารถจ่ายกระแสไฟฟ้า แต่ละพอร์ตได้ 40 มิลลิแอมป์ 3
2.1.1.7 สามารถจ่ายกระแสไฟฟ้าในพอร์ต 3.3 V จ่ายได้ 50 มิลลิแอมป์ 3
2.1.1.8 มีพื้นที่หน่วยความจำโปรแกรม 32 กิโลไบต์ 3
2.1.1.9 มีพื้นที่หน่วยความจำชั่วคราวแบบ SRAM 2 กิโลไบต์(KB) 3
2.1.1.10 มีพื้นที่หน่วยความจำถาวรแบบ EEPROM 1 กิโลไบต์ (KB) 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.1.1.11 ใช้ความถีสัญญาณนาฬิกา 16 เมกะเฮิรตซ์ (MHz) 3
2.1.2.1 ภาคจ่ายไฟฟ้า 4
2.1.2.2 หน่วยความจำ 4
2.1.2.3 พอร์ตอินพุต - เอาต์พุต 5
2.1.2.4 อนาล็อกอินพุต 5
2.1.2.5 การสื่อสาร 6
2.2 Motor Drive Shield dual L293D for arduino 6
2.3 DC 3V-6V 1:48 Gear Motor smart robot 7
2.4 ล้อยาง WHEEL SMART CAR CHASSIS 9
2.5เซ็นเซอร์วัดระยะทาง Ultrasonic Module HC-SR04 9
2.6 ถ่าน18650 11
2.7 รางถ่าน (กระบะถ่าน) 18650 2 ก้อน 13
2.8 แผ่นอะคริลิค 13
2.9สวิตช์ 14
2.10 GPS Module GY-NEO6MV2 15
2.12.1.1 ตำแหน่ง ดาวเทียม GPS ในอวกาศ อย่างน้อย 3 ดวง 15
2.12.1.2 ระยะห่างจาก ดาวเทียม GPS แต่ละดวง 15
2.12.2.1การได้มาซึ่งระยะห่างของอุปกรณ์รับGPSกบั ดาวเทียมGPSแต่ละดวง 16
2.11 สายไฟจั้มเปอร์ 18
2.12 Bluetooth Module HC-05 18
2.13 HMC5883L 19
บทที่ 3 การออกแบบและวิธีการดำเนินการ 22
3.1 การออกแบบ 22
3.2 ขั้นตอนการทำงานในส่วนของอุปกรณ์ 23
3.2.1 Arduino Uno 24
3.2.2 Motor Driver Shield 24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.2.3 TT Gear Motor 24
3.2.4 Ultrasonic Sensor 24
3.2.5 Bluetooth Module 24
3.2.6 GPS Module 24
3.2.7 Magnetometer Sensor Module 24
3.2.8 ตำแหน่งการวางอุปกรณ์ 25
3.3 ขั้นตอนการทำงานในส่วนของการแสดงผล 26
3.3.1 แอพพลิเคชั่น Blynk 26
3.3.2 ตำแหน่งปุ่มกด Blynk 30
3.3.3 การเขียนโปรแกรม 30
บทที่ 4 ผลการทดลอง 31
4.1 กล่าวนำ 31
4.2 ขั้นตอนการทดลอง 31
4.3.1 ขั้นตอน 1 31
4.3.2 ขั้นตอน 2 31
4.3.3 ขั้นตอน 3 31
4.3 ผลการทดลอง 32
4.4 อุปสรรคในการทดลอง 32
บทที่ 5 สรุปผลการทดลอง 33

บรรณานุกรม 35

ภาคผนวก 36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

2.1 บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3....4
2.2 Motor Drive Shield dual L293D....6
2.3 DC 3V-6V 1:48Gear Motor smart robot....7
2.4 สเปคแกนล้อมอเตอร์....8
2.5 รูปแบบชิ้นงาน....8
2.6 ล้อยาง....9
2.7 Ultrasonic Module HC-SR04....9
2.8 สัญญาณ timing diagram....10
2.9 การคำนวณนำไปใช้งาน....10
2.10 ถ่าน18650....11
2.11 โครงสร้างภายในของแบตเตอรี่ 18650....12
2.12รางถ่าน....13
2.13 แผ่นอะคริลิค....13
2.14 สัญลักษณ์วงจรไฟฟ้า....14
2.15 GPS Module GY-NEO6MV2....15
2.16 สายไฟจั้มเปอร์....18
2.17 HC-05....18
2.18 HMC5883L GY-271....19
2.19 haversine....20
2.20 สมการคำนวณ พิกัดละติจูดและลองจิจูด....20
2.21 สมการการคำนวณมุมแอซิมัท....21
2.22 Azimuth angle....21
3.1 Flow Chart Program....22
3.2 โครงสร้างการออกแบบและหลักการทำงาน....23
3.3 การเชื่อมต่ออุปกรณ์และการประกอบโครงรถ....23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

3.4 ตำแหน่งการติดตั้งอุปกรณ์ใต้ตัวรถ 25
3.5 ตำแหน่งการติดตั้งอุปกรณ์ใต้ตัวรถ 25
3.6 หน้าต่างโปรแกรม Blynk 27
3.7 แสดงหน้าจอการใช้งานของแอพพลิเคชั่น Blynk 27
3.8 หน้าต่างบล็อคหรือปุ่มที่ต้องการเลือก 28
3.9 โค้ดโปรแกรม Arduino 28
3.10 โปรแกรมที่พร้อมใช้งาน 30
4.1 รถสถานะพร้อมใช้งาน 33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันภาวะเศรษฐกิจโลก มีการพัฒนาเทคโนโลยีทางด้านอุตสาหกรรมมากมาย ทำให้เศรษฐกิจต้องพึ่งพาเทคโนโลยีที่ทันสมัยและเหมาะสมกับงานอุตสาหกรรมที่นำไปใช้หรือความต้องการการใช้งาน

CarTracker โดยใช้ arduino มีความสำคัญอย่างยิ่ง เป็นพื้นฐานส่วนหนึ่งในการเรียนการสอนการทำ CarTracker โดยใช้ arduino แต่สิ่งที่หน้าสนใจของโครงการนี้คือ การเข้ามาช่วยนักเรียนนักศึกษาเนื่องจากการเรียนการสอนในปัจจุบันมีการเรียนที่ใช้ความรู้ทางด้าน arduino เป็นหลัก ดังนั้นโครงการจึงสร้าง CarTracker โดยใช้ arduino การสร้างในทุกๆอย่าง มีการทำงานเป็นขั้นตอนและเป็นลำดับ ซึ่งถ้าหากข้ามขั้นตอนใดไปอาจทำให้การสร้าง CarTracker นั้นมาได้ไม่ตรงเป้าหมาย จึงได้มีการจัดทำ CarTracker โดยใช้ arduino เพื่อใช้ในการเรียนการสอนตามรายวิชาที่ต้องใช้ arduino เป็นส่วนใหญ่ เพื่อให้นักเรียนนักศึกษาเห็นอุปกรณ์และการทำงานของจริงเข้ามาทดแทนการเรียนการสอนแบบเดิมๆ

การทำงานของ CarTracker ใช้บอร์ด arduino ในการควบคุมวงจรและเขียนคำสั่งโปรแกรมไปยังส่วนต่างๆ จะทำการใช้งานต้องการเปิดสวิช์เปิดให้แบตเตอรี่ทำงานและจ่ายไฟเลี้ยงบอร์ด arduino ให้ทำงาน โดยมีเซ็นเซอร์เป็นส่วนที่รับข้อมูล ส่งข้อมูลไปยังบอร์ด arduino ทำให้ล้อรถขยับตามทิศทางที่เรากำหนดและบอกค่าพิกัดตำแหน่ง โดยใช้ GPS ระบุขึ้นใน google map เป็น CarTracker ที่สามารถติดตามคนใช้งานไปที่ต่างๆได้

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1.2.1 เพื่อใช้หลักการทำงานแบบไร้คนบังคับ เป็นแบบไร้สายผ่านแอพในการควบคุม

1.2.2 เพื่อเป็นสื่อกลางในการเรียนการสอนในวิชาที่ต้องใช้ arduino เป็นหลัก

1.2.3 เพื่อเป็นแนวทางในการสร้างสิ่งประดิษฐ์พื้นฐานในการใช้อุปกรณ์และโปรแกรมในการ CarTracker

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของปริญญานิพนธ์

โครงงานนี้เป็นการสร้าง CarTracker ซึ่งจำเป็นต้องศึกษาตั้งแต่ส่วนประกอบของ arduino อุปกรณ์ และการนำมาใช้งาน การใช้งานการเขียนโปรแกรม รวมทั้งการนำความรู้ที่ได้จากการเรียนที่ผ่านมามาประยุกต์ใช้ให้เกิดเป็นโครงงานดังกล่าว

1.3.1 สามารถควบคุมผ่านโทรศัพท์มือถือโดยใช้แอพพลิเคชั่นควบคุม

1.3.2 กำหนดพิกัดตำแหน่ง CarTracker

1.3.3 สามารถติดตามคนใช้งานด้วยเซ็นเซอร์

1.4 ประโยชน์คาดที่จะได้รับ

1.4.1 ได้อุปกรณ์ชิ้นงาน CarTracker

1.4.2 ได้ใช้ในการเรียนการสอนในห้องเรียน

1.4.3 ได้ศึกษาความรู้ความเข้าใจจากการทำโครงการ

1.5 ขั้นตอนการศึกษา

1.5.1 ศึกษาการเขียนโปรแกรมของบอร์ด Aruduino และอุปกรณ์ที่จะใช้งานกัน บอดร์ด arduino

1.5.2 ศึกษาเงื่อนไขการทำงานของโค้ดโปรแกรมและการติดตั้งของอุปกรณ์

1.5.3 ศึกษาข้อมูลตัว GPS ในการควบคุม

1.5.4 ศึกษาข้อมูลและปัญหาของอุปกรณ์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3

บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3 ได้รับความนิยมอีกบอร์ดหนึ่ง เนื่องจากมีราคา ไม่แพง ซึ่งส่วนใหญ่โปรเจคและไลบรารีต่าง ๆ ที่พัฒนาขึ้นมาถูกอ้างอิงกับบอร์ดนี้เป็นหลัก เพราะเป็น ขนาดที่เหมาะกับการเริ่มต้นการเรียนรู้ Arduino ซึ่งบอร์ด Arduino Uno ได้ถูกพัฒนาขึ้นมา ตั้งแต่ R2, R3 และมีรุ่นชิปไอซีเป็นแบบ SMD ในการเรียนรู้บอร์ดไมโครคอนโทรลเลอร์ Arduino เล่มนี้ใช้ เป็นบอร์ดรุ่น Arduino Uno R3 มีคุณสมบัติของบอร์ดดังนี้

2.1.1.1 ใช้ไอซีไมโครคอนโทรลเลอร์ ATmega328

2.1.1.2 ใช้แรงดันไฟฟ้าเลี้ยงไอซีไมโครคอนโทรลเลอร์ ATmega328 มีค่า 5 โวลต์

2.1.1.3 แรงดันไฟฟ้าป้อนที่บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3 อยู่ในช่วง 7 - 12โวลต์

2.1.1.4 มีพอร์ตดิจิตอลอินพุต/เอาต์พุต (Digital VO) จำนวน 14 พอร์ต (มี PWM output จำนวน 6 พอร์ต)

2.1.1.5 มีพอร์ตอนาล็อกอินพุต (Analog Input) จำนวน 6 พอร์ต

2.1.1.6 สามารถจ่ายกระแสไฟฟ้า แต่ละพอร์ตได้ 40 มิลลิแอมป์ (mA)

2.1.1.7 สามารถจ่ายกระแสไฟฟ้าในพอร์ต 3.3 V จ่ายได้ 50 มิลลิแอมป์ (mA)

2.1.1.8 มีพื้นที่หน่วยความจำโปรแกรม 32 กิโลไบต์ (KB)

2.1.1.9 มีพื้นที่หน่วยความจำชั่วคราวแบบ SRAM 2 กิโลไบต์ (KB)

2.1.1.10 มีพื้นที่หน่วยความจำถาวรแบบ EEPROM 1 กิโลไบต์ (KB)

2.1.1.11 ใช้ความถีสัญญาณนาฬิกา 16 เมกะเฮิรตซ์ (MHz)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.1 บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3

2.1.2.1 ภาคจ่ายไฟฟ้า

บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3 สามารถใช้พลังงานไฟฟ้าจากพอร์ต USB ของคอมพิวเตอร์ หรือแหล่งพลังงานไฟฟ้าจากภายนอกโดยบอร์ดสามารถเลือกแหล่งจ่ายไฟฟ้าได้โดย อัตโนมัติ ในส่วนของแหล่งพลังงานไฟฟ้าจากภายนอก สามารถใช้ได้ทั้งแบบไฟฟ้ากระแสสลับและ ไฟฟ้ากระแสตรงจากอะแดปเตอร์ หรือจากแบตเตอรี่โดยมีขั้วไฟฟ้าของอะแดปเตอร์สามารถเชื่อมต่อ ด้วยการเสียบปลักขนาด 2.1 มม. เข้ากับแจ็คพาวเวอร์ของบอร์ด ช่วงแรงดันของแหล่งจ่ายไฟฟ้าที่ แนะนำควรมีค่าอยู่ในช่วง 7 - 12 โวลต์ แต่ถ้าใช้แรงดันไฟฟ้ามากกว่า 12 โวลต์ ส่งผลให้ไอซีควบคุม แรงดันไฟฟ้าร้อนมากเกินไปและเกิดความเสียหายต่อบอร์ดได้ ขาพาวเวอร์ซัพพลาย มีดังนี้

- Vin เป็นขารับแรงดันไฟฟ้าเลียงบอร์ด Arduino จากแหล่งจ่ายไฟฟ้าภายนอก

- 5 V เป็นขาจ่ายแรงดันไฟฟ้า 5 โวลต์ ที่ได้จากแรงดันจาก Vin ผ่านวงจรเร็กกูเลเตอร์ภายในบอร์ด หรือจากแรงดันไฟฟ้าที่พอร์ต USB

- 3.3V เป็นขาจ่ายแรงดันไฟฟ้า 3.3 โวลต์ ที่สร้างขึ้นโดยวงจรเร็กกเลเตอร์ภายในบอร์ดจ่ายกระแสสูงสุดคือ 50 มิลลิแอมป์

- GND เป็นขากราวนด์

2.1.2.2 หน่วยความจำ

ไมโครคอนโทรลเลอร์ ATmega328 มีหน่วยความจำแบบแฟลชสำหรับการจัดเก็บโปรแกรม ขนาด 32 กิโลไบต์ (มีหน่วยความจำใช้สำหรับการบูต ขนาด 0.5 กิโลไบต์) มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำชั่วคราวแบบสแตติกแรม (SRAM) ขนาด 2 กิโลไบต์ และมีหน่วยความจำถาวรแบบอีอีพร็อม (EEPROM)ขนาด 1 กิโลไบต์

2.1.2.3 พอร์ตอินพุต - เอาต์พุต

บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3 มีพอร์ตดิจิตอลทั้งหมด 14 ขา สามารถเป็นได้ทั้งพอร์ตอินพุตหรือเอาต์พุต โดยใช้ฟังก์ชัน pinMode (), digitalWrite) และ digitalRead () แต่ละขาทำงานที่แรงดัน 5 โวลต์ สามารถจ่ายหรือรับกระแสไฟฟ้าได้สูงสุด 40 มิลลิแอมป์ และมียังมีหน้าที่พิเศษ ดังนี้ ตัวต้านทานต่อแบบพูลอัปอยู่ภายในมีค่าความต้านทาน 20 - 50 กิโลโอห์ม นอกจากนี้แล้ว บางพอร์ต

- พอร์ต O เป็นขา RX ใช้เป็นพอร์ตรับสัญญาณสื่อสารแบบอนุกรม

- พอร์ต 1 เป็นขา TX ใช้เป็นพอร์ตส่งสัญญาณสื่อสารแบบอนุกรม

- พอร์ต 2 และ 3 เป็นพอร์ตรับสัญญาณอินเตอร์รัพท์ภายนอก (Interrupts) พอร์ตเหล่านี้สามารถกำหนดค่าให้รับสัญญาณขัดจังหวะได้ทั้งแบบลอจิกสูง ลอจิกต่ำ หรือแบบอื่น ๆ

- พอร์ต 3, 5, 6, 9, 10 และ 11 เป็นพอร์ตส่งสัญญาณเอาต์พุตแบบ PWM ขนาด 8 บิต

- พอร์ต 10, 11, 12 และ 13 เป็นพอร์ตสื่อสารแบบ SPI

- พอร์ต 13 เป็นพอร์ตควบคุมแอลอีดีที่ติดตั้งบนบอร์ด เมื่อขา 13 จ่ายเอาต์พุตลอจิก “1” ทำให้แอลอีดีติดสว่าง และเมื่อจ่ายลอจิก “0” ทำให้แอลอีดีดับ

2.1.2.4 อนาล็อกอินพุต

บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3 มีพอร์ตอนาล็อกอินพุต 6 ขา คือ ขา AO - ขา A5 ซึ่งแต่ช่องมีความละเอียดขนาด 10 บิต แบ่งระดับความแตกต่างได้ 1,024 ค่า โดยเริ่มต้น จากระดับแรงดัน 0 โวลต์ จนถึงระดับ 5 โวลต์ และสามารถเปลี่ยนระดับแรงดันอ้างอิงได้โดยใช้ แรงดันอ้างอิงจากภายนอกป้อนที่ขา AREF ร่วมกับฟังก์ชัน analogReference() นอกจากนี้ยังมี บางขาที่มีหน้าที่พิเศษ ดังนี้

- พอร์ต A4 (SDA) และพอร์ต A5 (SCL) เป็นพอร์ตสื่อสารแบบ IRC

- พอร์ต Aref แรงดันอ้างอิงสำหรับอินพุตอนาล็อก ใช้งานร่วมกับฟังก์ชัน analog Reference ()

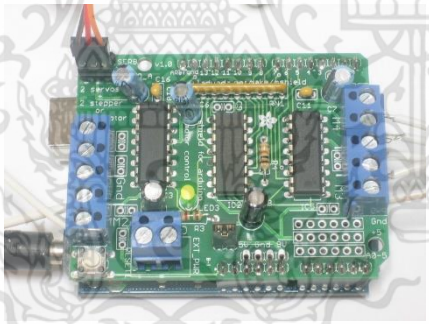
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.5 การสื่อสาร

บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno R3 มีพอร์ตสื่อสารเพื่อเชื่อมต่อกับคอมพิวเตอร์ หรือบอร์ด Arduino อื่นๆ หรือไมโครคอนโทรลเลอร์เบอร์อื่น ๆ หลายรูปแบบ ตามความสามารถของ ไมโครคอนโทรลเลอร์ ATmega328 ที่มีพอร์ตสื่อสารอนุกรมแบบ UART ที่พอร์ตดิจิตอล ขา 0 (R และพอร์ตดิจิตอล ขา 1 (TX) ช่องทางการสื่อสารแบบอนุกรมยังเชื่อมโยงผ่านพอร์ต USB และยัง ปรากฏเป็นพอร์ต COM เสมือนซอฟต์แวร์บนเครื่องคอมพิวเตอร์ ด้วยเฟิร์มแวร์ 8U2 คอมพิวเตอร์ สามารถเชื่อมต่อได้โดยใช้ไดรเวอร์ USB มาตรฐาน และไม่ต้องใช้ไดรเวอร์ภายนอกแต่อย่างไร

ซอฟต์แวร์ Arduino แสดงผลการสื่อสารผ่านพอร์ตแบบอนุกรมทางพอร์ต USB ระหว่างบอร์ด Arduino กับคอมพิวเตอร์ผ่านขา RX และ TX ซึ่งทำให้ไฟ LED กะพริบเมื่อข้อมูลถูกส่งผ่านทางพอร์ต USB

2.2 Motor Drive Shield dual L293D for arduino



รูปที่ 2.2 Motor Drive Shield dual L293D

Arduino เป็นจุดเริ่มต้นที่ยอดเยี่ยมสำหรับอุปกรณ์อิเล็กทรอนิกส์และด้วยเกราะป้องกันมอเตอร์ก็สามารถเป็นแพลตฟอร์มที่เป็นระเบียบเรียบร้อยสำหรับหุ่นยนต์และเมคคาทรอนิกส์ นี่คือการออกแบบสำหรับเกราะป้องกันมอเตอร์ที่มีคุณสมบัติครบถ้วนซึ่งจะสามารถขับเคลื่อนโครงการที่มีความซับซ้อนง่ายถึงปานกลางได้มากมาย

2 การเชื่อมต่อสำหรับเซอร์โว 'งานอดิเรก' 5V ที่ เชื่อมต่อกับตัวจับเวลาเฉพาะความละเอียดสูงของ Arduino - ไม่กระวนกระวายใจ!

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มอเตอร์กระแสตรงแบบสองทิศทางสูงสุด 4ตัวพร้อมการเลือกความเร็ว 8 บิตแต่ละตัว (ดังนั้นความละเอียดประมาณ 0.5%)

มอเตอร์สเต็ปสูงสุด 2 ตัว (ยูนิโพลาร์หรือไบโพลาร์) พร้อมขดลวดเดี่ยวขดลวดคู่อินเตอร์ลีฟหรือไมโครสเต็ปปิ้ง

4 H-Bridges: ชิปเซ็ต L293D ให้ 0.6A ต่อบริดจ์ (สูงสุด 1.2A) พร้อมระบบป้องกันการปิดด้วยความร้อน 4.5V ถึง 25V

ดึงตัวต้านทานลงเพื่อป้องกันไม่ให้มอเตอร์ปิดใช้งานระหว่างการเปิดเครื่อง

ขั้วต่อเทอร์มินัลบล็อกขนาดใหญ่เพื่อต่อสายไฟ (10-22AWG) และพลังงานได้อย่างง่ายดาย

ปุ่มรีเซ็ต Arduino ปรากฏขึ้นด้านบน

แผงขั้วต่อ 2 พินเพื่อเชื่อมต่อแหล่งจ่ายไฟภายนอกสำหรับอุปกรณ์ลอจิก / มอเตอร์แยกกัน

ผ่านการทดสอบกับ Mega, Diecimila และ Duemilanove

2.3 DC 3V-6V 1:48 Gear Motor smart robot

รูปที่ 2.3 DC 3V-6V 1:48 Gear Motor smart robot

มอเตอร์กระแสตรงและชุดล้อสำหรับสร้างหุ่นยนต์! มอเตอร์เหล่านี้มีน้ำหนักเบาแรงบิดสูงและรอบต่อนาทีต่ำ สามารถปีนขึ้นเนินและมีแรงฉุดที่ดีเยี่ยมรวมทั้งคุณสามารถติดล้อที่ด้านใดด้านหนึ่งของมอเตอร์ด้วยเพลาขับสองด้าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Package Content

1 Geared Motor

Specifications

Motor Voltage: 3 - 12 V

Motor Current: 70 mA (typical) - 250 mA (max)

Speed: up to 170 RPM

Torque: up to 0.8 Kg

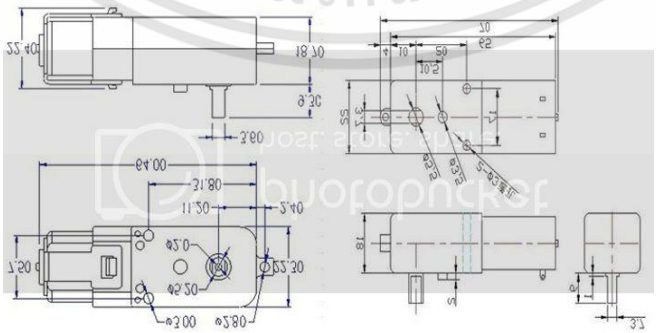
Gear Ration: 1:48

Wheel Diameter: 66 mm

Wheel Width: 27 mm

Voltage	DC 3V	DC 5V	DC 6V
Current	100 MA	100MA	120MA
Reduction rate	48:1		
RPM (With tire)	120	208	240
Tire Diameter	66mm		
Car Speed(M/minute)	20	39	48
Motor Weight (g)	50		
Motor Size	70mm*22mm*18mm		
Noise	<65dB		

รูปที่ 2.4 สเปคแกนล้อมอเตอร์



รูปที่ 2.5 รูปแบบชิ้นงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ล้อยาง WHEEL SMART CAR CHASSIS

รูปที่ 2.6 ล้อยาง

สามารถต่อร่วมกับ TT มอเตอร์ได้ เป็นล้อยางที่น้ำหนักเบา ง่ายต่อการติดตั้ง

ลักษณะการทำงาน: มียางในเส้นผ่านศูนย์กลาง: 68 มม. ความกว้าง: 26 มม. รูตรงกลาง: 5.3 x 3.66 มม. (สองด้านเป็นรูปครึ่งวงกลม) น้ำหนัก: 50g

แพคเกจรวม: 2 x ล้อรถหุ่นยนต์อัจฉริยะ

2.5 เซ็นเซอร์วัดระยะทาง Ultrasonic Module HC-SR04

รูปที่ 2.7 Ultrasonic Module HC-SR04

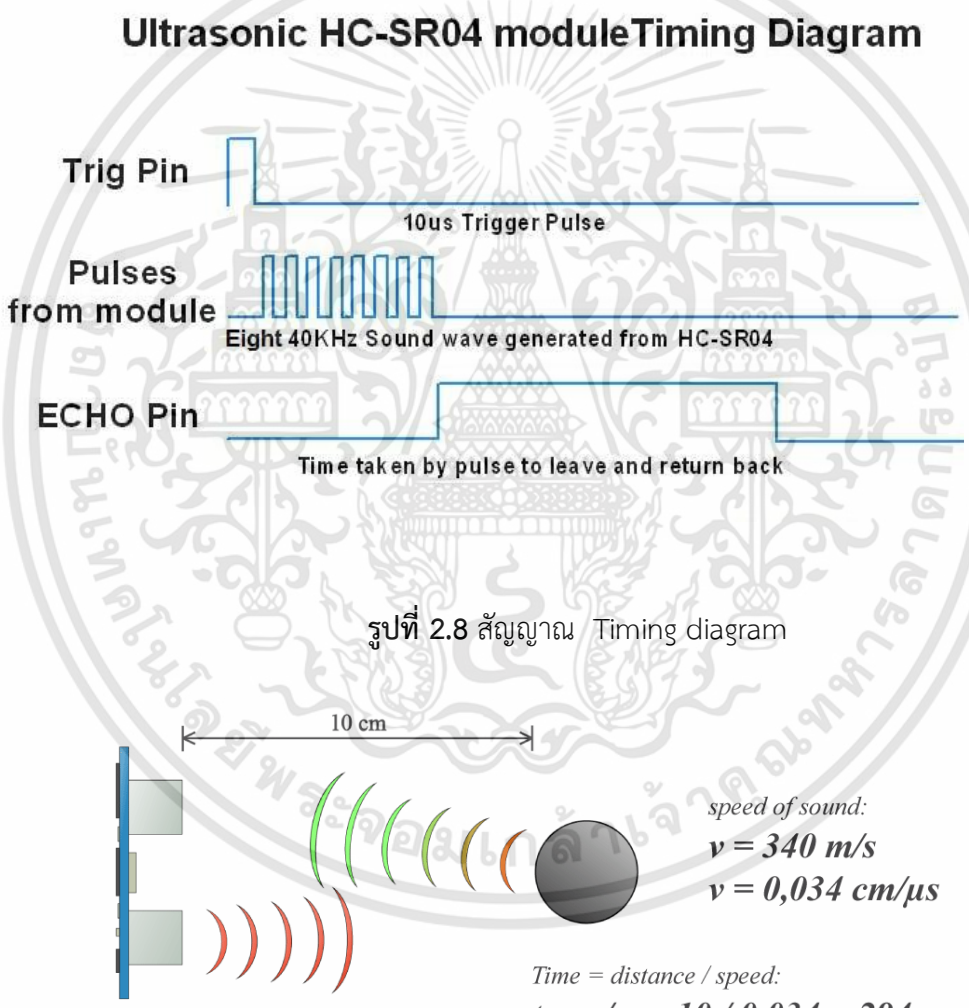
โมดูลเซนเซอร์วัดระยะทางแบบ Ultrasonic รุ่น HC-SR04P ปรับปรุงจาก HC-SR04 Ultrasonic Sensor Module รุ่นเดิมดังนี้ ใช้ไฟเลี้ยงได้กว้างขึ้น 3-5.5V (เดิม 5V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

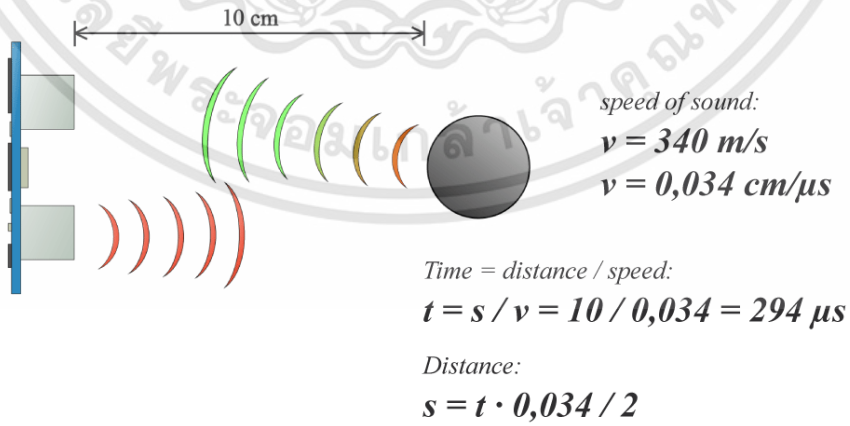
Ultrasonic ranging module HC-SR04P

โมดูลอัลตร้าโซนิคนี้เป็นอุปกรณ์ใช้วัดระยะทางโดยไม่ต้องมีการสัมผัสกับตำแหน่งที่ต้องการวัด วัดได้ตั้งแต่ 2 cm ถึง 450 cm โดยส่งสัญญาณอัลตร้าโซนิคความถี่ 40 kHz ไปที่วัตถุที่ต้องการวัดและรับสัญญาณที่สะท้อนกลับมา พร้อมทั้งจับเวลาเพื่อนำมาใช้ในการคำนวณระยะทาง

ในการปล่อย ultrasound เราต้องตั้งค่าขา Trig ให้อยู่ในสถานะการทำงานเป็นเวลา 10 µs ที่จะส่งultrasonic burst 8 รอบซึ่งจะเดินทางด้วยความเร็วของเสียงและเมื่อได้รับการสะท้อนเขาไปใน Echo จะแสดงเวลาเป็นไมโครวินาทีที่คลื่นเสียงเดินทาง



รูปที่ 2.8 สัญญาณ Timing diagram



รูปที่ 2.9 การคำนวณนำไปใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเช่น หากวัตถุอยู่ห่างจากเซ็นเซอร์ 10 ซม. และความเร็วของเสียงคือ 340 ม./วินาที หรือ 0.034 ซม./µs คลื่นเสียงจะต้องเดินทางประมาณ 294 น วินาที แต่สิ่งที่คุณจะได้รับจาก ขา Echo จะเป็น 2 เท่าของตัวเลขนั้น เนื่องจากคลื่นเสียงจะต้องเคลื่อนที่ไปข้างหน้าและย้อนกลับ ดังนั้นเพื่อให้ได้ระยะทางเป็น CM เราจำเป็นต้องคูณค่าเวลาเดินทางที่ได้รับจากหมุดสะท้อนด้วย 0.034 แล้วหารด้วย 2

2.6 ถ่าน18650

รูปที่ 2.10 ถ่าน18650

แบตเตอรี่ 18650 ถูกเรียกตามเส้นผ่านศูนย์กลางและความยาวของตัว (18mm x 65mm) ด้านในตัวแบตเตอรี่เป็น cell ชนิด Lithium – Ion แบตเตอรรี่ชนิดนี้ถูกใช้มาค่อนข้างนานแล้ว เพียงแต่อยู่ในรูปแบบของ Battery Pack จึงไม่ค่อยคุ้นเคยกันมากนัก แต่ในปัจจุบันเนื่องจากแบตเตอรี่ 18650 สามารถจ่ายกระแสได้สูงกว่าแบตเตอรี่ขนาด AA และความจุที่สูงกว่ามาก จึงเป็นที่นิยมแพร่งหลายในการใช้กับอุปกรณ์อิเล็กทรอนิกส์ต่างๆ เช่น คอมพิวเตอร์โน๊ตบุ๊ค ไฟฉายแรงสูง สว่านไฟฟ้า หรือแม้แต่ ชาร์ตแบตสำรอง หรือ Power Bank นั่นเอง แต่ด้วยคุณสมบัติเฉพาะของแบตเตอรี่ชนิด Lithium นั้นไม่เหมือนแบตเตอรี่ธรรมดาเช่น Ni-mh หรือแบตเตอรี่แห้ง ตัวแบตเตอรี่เองต้องจึงมีการออกแบบระบบป้องกันให้มีความปลอดภัยมากขึ้น ซึ่งแต่ละ cell ปรกติจะมีชุดป้องกันด้วยกัน 3 ชั้น โดยระบบป้องกันแบบ PCT และ CID ตามปรกติจะมีอยู่ใน cell อยู่แล้ว ส่วน Protected PCB นั้นขึ้นอยู่กับผู้ผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.11 โครงสร้างภายในของแบตเตอรี่ 18650

2.8.1. PCT จะเป็นตัวป้องกันความร้อนของ Cell เกินพิกัด และตัดการทำงานของแบตเตอรรี่ลง โดยจะสามารถกลับมาใช้งานได้เมื่ออุณหภูมิเข้าสู่สภาะวะปรกติ

2.8.2. CID นั้นจะเป็นวาล์วป้องกันความดันภายใน Cell เกินพิกัดจนอาจนำไปสู่การระเบิดได้ โดยวาล์วตัวนี้จะทำหน้าที่ตัดการทำงานของ Cell ถาวร ไม่สามารถคืนสภาะกลับมาใช้ใหม่ได้อีก หากสังเกตที่ขั้วของแบตเตอรี่จะพบรูเล็กๆ ที่ถูกออกแบบไว้สำหรับระบายแก๊สหากมีแรงดันผิดปรกติภายใน cell นั่นเอง

2.8.3.Protected PCB หรือเรียกกันว่า 18650 แบบมีวงจร ซึ่งเป็นวงจรอิเล็กทรอนิกส์ขนาดเล็กที่ฝังไว้ภายในตูดของตัวแบตเตอรี่ ทำหน้าที่คอยป้องกันการใช้กระแสเกิน (Over Current) ป้องกันแรงดันชาร์ตเกิน (Over Charge Voltage) และป้องกันการใช้ไฟในระดับโวลต์ที่ต่ำกว่ากำหนด (Over Discharge) โดยภายในจะมี IC ที่คอย ตรวจสอบอยู่ตลอดเวลาโดย IC ถูกออกแบบให้มีการกินกระแสน้อยมากๆ ในระดับไมโครแอมป์ ซึ่งแทบจะไม่ส่งผลกับปริมาณแบตเตอรี่เลย

ระดับแรงดันใช้งานปรกติของแบตเตอรี่ชนิด lithium จะอยู่ที่ 3.7V โดยแรงดันที่ชาร์จเต็มจะอยู่ที่ 4.2V เนื่องจากคุณสมบัติเฉพาะของ Cell ชนิด lithium นั้น หากมีดึงกระแสไฟจากแบตเตอรี่จนแรงดันต่ำกว่า 2.5V นั้นจะทำให้ cell เสียหายถาวร ไม่สามารถนำกลับมาใช้ใหม่ได้ (เครื่องชาร์จจะไม่ยอมชาร์ตหากแรงดันใน Cell ต่ำกว่าที่กำหนด) ฉะนั้น วงจรป้องกัน Protected PCB จึงเป็นสิ่งสำคัญอย่างมากในการใช้งานร่วมกับอุปกร์ณทั่วไป โดยผู้ใช้ไม่ต้องกังวลว่าจะใช้งานจากแบตเตอรี่หมดจน cell พัง อีกทั้งหากมีการดึงกระแสเกินกำหนดหรือมีการใช้แรงดันชาร์ตเกินพิกัด วงจรจะทำการตัดการทำงานอัตโนมัติ ส่วน 18650 แบบไม่มีวงจร นิยมใช้กับอุปกรณ์อิเล็กทรอนิกส์ที่ถูกออกแบบมาเฉพาะซึ่งมีวงจรควบคุมอยู่ภายนอกแล้ว เช่น วงจร BMS หรือ PCM ซึ่งมีหลักการทำงานคล้ายกัน จึงไม่จำเป็นต้องมีวงจรป้องกันภายในตัว cell ทำให้ประหยัดต้นทุนในการผลิต มักพบเห็นใน Battery Pack

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 รางถ่าน (กระบะถ่าน) 18650 2 ก้อน

รูปที่ 2.12 รางถ่าน

สำหรับ ใส่battery 18650 ทำการส่งไฟเลี้ยงไปยังส่วนต่างๆ

2.8 แผ่นอะคริลิค

รูปที่ 2.13 แผ่นอะคริลิค

แผ่นอะคริลิค “Acrylic” เป็นแผ่นพลาสติกเรียบจำพวก Thermoplastic ซึ่งผลิตขึ้นจากน้ำยา MMA (Methyl Methacrylate) นำไปเข้าระบบหล่อแบบ (Casting System) ซึ่งมีลักษณะเด่นหมายถึงเมื่อได้รับความร้อนสูงจะอ่อนตัวลง สามารถดัดหรือขึ้นรูปเป็นแบบต่างๆได้ รวมทั้งเมื่อเย็นตัวลงจะแข็งและก็ทรงสภาพไว้ มีน้ำหนักเบา สามารถสลัก ตัดเลเซอร์ พ่นสี ระบาย พิมพ์สกรีน พิมพ์แสงอัลตราไวโอเลต ปั๊มทองนอก เป็นรูปหรือลวดลายต่างๆได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรรมวิธีการผลิต ของ แผ่นอะคริลิคใน 2 ระบบ ไม่เหมือนกันในด้านของวัตถุดิบที่ใช้เพื่อการผลิต โดยในระบบ Casting จะใช้น้ำยา MMA เป็นวัตถุดิบสำหรับในการผลิต ส่วนในระบบ Extrusion ใช้เม็ด PMMA สำหรับการผลิต สำหรับคุณลักษณะจะใกล้เคียงกัน เพราะว่ามีวัตถุดิบพื้นฐานประเภทเดียวกัน ก็แค่เมื่อผ่าน process ที่แตกต่างก็เลยทำให้มีข้อกำหนดการใช้แรงงานแตกต่างกันซึ่ง ขึ้นกับการนำไปใช้งานในรูปแบบต่างๆด้วย

แนวทางการพับแผ่นอะคริลิคเป็นมุมต่างๆเป็นนำอะคริลิคไปให้ความร้อนตามแนวเส้นลวดความร้อน เมื่ออะคริลิคนิ่มตัว ก็เลยกระทำการพับให้ได้มุมดังที่อยากได้

ลักษณะเด่น อีกอย่างหนึ่งของแผ่นอะคริลิคเป็นสามารถทนแรงชนได้ดีมากว่ากระจก โดยความหนาของแผ่นจะเป็นเหตุที่แปรผันโดยตรงกับการทนแรงชน ขนาดความครึ้มของแผ่นอะคริลิคมีตั้งแต่ 2 มม. – 40 มม. มีขนาด 4*6 รวมทั้ง 4*8 ฟุต สามารถเอามาสร้างเป็นข้าวของต่างๆได้หลากหลาย เป็นต้นว่า โล่รางวัล, โพเดี่ยม, กรอบรูป, ชั้นที่มีไว้สำหรับวางสิ่งของ,ที่โชว์ผลิตภัณฑ์, ป้ายที่ใช้สำหรับโฆษณา, อื่นๆอีกมากมาย

ข้อดีของแผ่นอะคริลิค อยู่ที่ความแข็งแรง ทนต่อความร้อนรวมทั้งแสงอาทิตย์ เมื่อเทียบกับไม้ ซึ่งมีความจำกัดเมื่อเอามาวางที่โล่งแจ้งเป็นระยะเวลาที่ยาวนาน สีของไม้ จะแปลงรวมทั้งผุพังได้ง่าย ด้วยเหตุผลดังกล่าว เครื่องเรือนจากอ่างอาบน้ำก็เลยแข็งแรง ทน สีแจ่มใส อายุการใช้งานนานโดยประมาณ 10 ปี แล้วก็รองรับน้ำหนักผู้นั่งได้ 3-4 คน

2.9 สวิตช์

อุปกรณ์ที่ทำหน้าที่ควบคุมการไหลของกระแสไฟฟ้าภายในวงจร หรือกล่าวง่าย ๆ คือ อุปกรณ์เปิด ปิดกระแสไฟฟ้าภายในวงจรไฟฟ้า โดยใช้สัญลักษณ์ดังรูป

รูปที่ 2.14 สัญลักษณ์วงจรไฟฟ้า

สวิตช์ที่ใช้ในงานอิเล็กทรอนิกส์มีหลายชนิด เช่น สวิตช์เลื่อน สวิตช์กระดก สวิตช์หมุน สวิตช์กด สวิตช์ไมโคร สวิตช์กุญแจ ฯลฯ

สวิตช์เลื่อน เป็นสวิตช์ชนิดหนึ่งที่ใช้เปิด ปิด การทำงานของอุปกรณ์ ใช้งานโดยการเลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมตัดต่อสวิตช์ ทำได้โดยผลักเลื่อนสวิตช์ขึ้นบนหรือลงล่าง การเลื่อนสวิตช์ขึ้นบนเป็นการต่อ (ON) การเลื่อนสวิตช์ลงล่างเป็นการตัด (OFF) นิยมใช้เป็นอุปกรณ์เปิด ปิด สิ่งของประเภทของเล่นเด็ก และเครื่องใช้ต่างๆ เช่น นาฬิกาปลุก ไฟฉาย

2.10 GPS Module GY-NEO6MV2

รูปที่ 2.15 GPS Module GY-NEO6MV2

หลักการทำงานของ GPS

หลักการของเครื่อง GPS คือ การคำนวณระยะทางระหว่างดาวเทียมกับอุปกรณ์รับ GPS โดยจะต้องทราบตำแหน่งของดาวเทียมแต่ละดวง ประกอบกับได้ระยะทางจากดาวเทียม 3 ดวง ขึ้นไปแล้ว อุปกรณ์ GPS ก็จะสามารถคำนวน หาจุดตัดกันของผิวทรงกลม ของระยะทางของดาวเทียม GPS แต่ละดวงได้

ดังนั้น ในทางทฤษฎี สิ่งที่อุปกรณ์ GPS จำเป็นต้องทราบในการคำนวนหาตำแหน่งแต่ละครั้ง คือ

2.12.1.1 ตำแหน่ง ดาวเทียม GPS ในอวกาศ อย่างน้อย 3 ดวง

2.12.1.2 ระยะห่างจาก ดาวเทียม GPS แต่ละดวง

โดยการจะได้มาซึ่ง ข้อมูลทั้ง 2แบบ ในทางปฏิบัติ คือ

1.การได้มา ซึ่ง ตำแหน่งดาวเทียม GPS ในอวกาศ

การได้มา ซึ่งตำแหน่งดาวเทียม GPS ในอวกาศ จะต้องได้มีข้อมูลประกอบ 2 ตัว คือ

a. ข้อมูลวงโคจร : จะทำให้อุปกรณ์ GPS ทราบว่า เส้นทางการเดินทางของดาวเทียม GPS แต่ละดวงจะอยู่ ณ ตำแหน่งใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

b. เวลาปัจจุบัน : ซึ่งเมื่ออุปกณ์ GPS ทราบ เวลาปัจจุบัน แล้ว ก็จะใช้เวลาปัจจุบัน ไปคำนวนหา ตำแหน่ง ของดาวเทียม GPS จากข้อมูลวงโคจรได้

ดังนั้น เมื่ออุปกรณ์รับ GPS ทราบ ข้อมูลวงโคจร ดาวเทียม GPS และเวลาปัจจุบัน อุปกรณ์รับ GPS ก็จะทราบตำแหน่ง ดาวเทียมในอวกาศได้ ซึ่งข้อมูลทั้งหมด จะได้มาจากสัญญาณดาวเทียมที่ อุปกรณ์รับ GPS ตัวนั้นรับได้

2.12.2.1 การได้มา ซึ่ง ระยะห่างของอุปกรณ์รับ GPS กับ ดาวเทียม GPS แต่ละดวง

เนื่องจาก การเดินทางของคลื่นสัญญาณ GPS นั้น จะเคลื่อนที่ด้วยความเร็วคงที่(vคงที่) คือ ความเร็วแสง (186,000ไมล์ต่อวินาที) ซึ่งเมื่อเป็นดังนั้น ถ้าอุปกรณ์รับ GPS รู้ระยะเวลา(t) ที่สัญญาณ ใช้ในการเดินทางจาก ดาวเทียม GPS มายังอุปกรณ์รับ GPS ก็จะสามารถคำนวนระยะทางระหว่าง ดาวเทียม GPS กับ อุปกรณ์รับ GPS ได้ จากสูตร

ความเร็ว X เวลา = ระยะทาง

ซึ่งเมื่อเราทราบระยะห่างของดาวเทียมกับอุปกรณ์ GPS มากเท่าไร เราก็จะหาจุดของผิวทรง กลม ทำให้อุปกรณ์ GPS สามารถทราบว่าตัวเองอยู่นะจุดใดบนพื้นโลกได้ เช่น

ดาวเทียม GPS 1 : ลอยอยู่ ณ จุดหนึ่งในอวกาศ ซึ่งเรารู้ตำแหน่ง จากข้อมูลวงโคจร GPS และ เวลาปัจจุบัน ระยะเวลาในการส่งสัญญาณจากดาวเทียมดวง GPS 1 ถึงเครื่องรับ GPS คือ 0.10 วินาที ระยะทางระหว่างดาวเทียมกับ GPS 1 คือ 18,600 ไมล์ (186,000 ไมล์ต่อวินาที X 0.10 วินาที = 18,600 ไมล์)

ดังนั้น ตำแหน่งปัจจุบัน ของเครื่องรับ GPS ก็จะสามารถเป็นจุดใดๆ ก็ได้ บนผิวทรงกลมที่มีรัศมี 18,600 ไมล์

รูปโลก โดน สัมผัสด้วยทรงกลม ใส

ดาวเทียม GPS 2 : ระยะเวลาในการส่งสัญญาณจากดาวเทียมดวง GPS 2 ถึงเครื่องรับ GPS คือ 0.08 วินาที ระยะทางระหว่างดาวเทียมกับ GPS 2 คือ 13,200 ไมล์ (186,000 ไมล์ต่อวินาที X 0.08 วินาที = 13,200 ไมล์)

ดังนั้น ตำแหน่งปัจจุบัน ของเครื่องรับ GPS ก็จะสามารถเป็นจุดใดๆ ก็ได้ บนเส้นรอบวงที่เป็น การตัดกันของ ทรงกลมรัศมี 18,600ไมล์ ของดาวเทียม GPS 1 กับ ทรงกลมรัศมี 13,200ไมล์ ของ ดาวเทียม GPS 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปโลก โดน สัมผัสด้วยทรงกลม ใส 2 วง

ดาวเทียม GPS 3 : ระยะเวลาในการส่งสัญญาณจากดาวเทียมดวง GPS 3 ถึงเครื่องรับ GPS คือ 0.06 วินาที ระยะทางระหว่างดาวเทียมกับ GPS 3 คือ 11,160 ไมล์ (186,000 ไมล์ต่อวินาที X 0.06 วินาที = 11,160 ไมล์)

ดังนั้น ตำแหน่งปัจจุบัน ของเครื่องรับ GPS ก็จะสามารถเป็นได้แค่ 2 จุด ที่เกิดจากจุดตัดของ ผิวทรงกลมรัศมี 18,600ไมล์ ของดาวเทียม GPS 1 กับ ผิวทรงกลมรัศมี 13,200ไมล์ ของดาวเทียม GPS 2 และ ผิวทรงกลมรัศมี 11,160 ไมล์ ของดาวเทียม GPS3

รูปโลก โดน สัมผัสด้วยทรงกลม ใส 3 วง

ดังนั้น หากอุปกรณ์ GPS ยิ่งสามารถรับสัญญาณจากดาวเทียม GPS มากดวงเท่าไร ก็จะยิ่งสามารถ ระบุตำแหน่งได้แม่นยำยิ่งขึ้น

ในกรณี ที่อุปกรณ์รับ GPS สามารถรับสัญญาณ GPS ได้จากดาวเทียม GPS เพียง 3 ดวง อุปกรณ์รับ GPS จะมีความสามารถในการประมาณตำแหน่งบนพื้นโลกได้ และจะตัดจุดที่ไม่ใช่ตำแหน่งบนพื้นโลกทิ้งไป ทำให้เหลือเพียงตำแหน่งแหน่งเดียวที่เป็นไปได้

จะเห็นได้ว่าจะเหลือตำแหน่งอยู่ 2 จุดที่บริเวณวงกลมทั้ง 3 ตัดกันคือตำแหน่งที่ อยู่ในอวกาศซึ่งแน่นอนว่าเราไม่สามารถไปอยู่ในอวกาศได้ตำแหน่งนี้จะถูกตัดทิ้งอัตโนมัติ โดยเครื่อง GPS อีกตำแหน่งคือตำแหน่งบนพื้นโลกซึ่งเป็นตำแหน่งที่เรายืนถือเครื่อง GPS อยู่นั้นเองซึ่งความถูกต้องแม่นยำของตำแหน่งก็ขึ้นกับจำนวนดาวเทียมที่สามารถรับ สัญญาณ ได้ในขณะนั้นหากมีมากกว่า 3 ดวงก็จะละเอียดมากขึ้น และก็ขึ้นกับเครื่อง GPS ด้วย หากเป็นเครื่องที่มีราคาแพง (ซึ่งมักใช้เฉพาะงาน) ก็จะมีความถูกต้องแม่นยำมากขึ้น

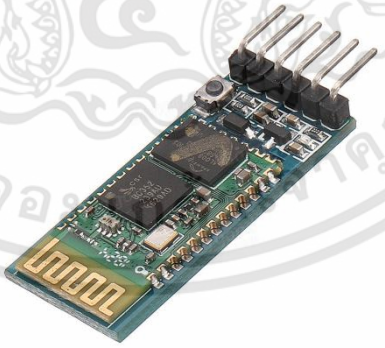
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 สายไฟจั้มเปอร์

รูปที่ 2.16 สายไฟจั้มเปอร์

สายไฟจัมเปอร์ คือ สายไฟ หรือ เทอร์มินอล ที่เชื่อมต่อระหว่างวงจรอิเล็กทรอนิกส์ระยะไกลและแผง อุปกรณ์ควบคุมเรียกอีกอย่างว่าข้ามเส้นซึ่งแตกต่างจากการ อุปกรณ์สายไฟ ธรรมดาเป็นการสร้างทางลัดที่สามารถกระโดดบนวงจรได้ชั่วคราวด้วยการใส่ สายไฟจัมเปอร์ ทำให้สามารถหยุดการทำงานของวงจรปกติและเปิดใช้งานวงจรที่ไม่ได้ใช้งานเป็นประจำใช้สำหรับการทดสอบการ งานซ่อมบำรุง เชื่อมต่อชั่วคราวใน เคส ฉุกเฉินเมื่อแบ่งวงจรตาม สเปค ผลิตภัณฑ์ ฯลฯมีสาย ไวนิล ทนความร้อน ยืดหยุ่น สายชุบสาย ฟลูออโรโพลิเมอร์ ฯลฯจัมเปอร์โลหะ รูปทรงตัว U เป็น ผลิตภัณฑ์ ที่ช่วยเพิ่มประสิทธิภาพในการทำงานเมื่อใช้งานในระยะห่างเดียวกันระหว่างจุดสองจุดที่เชื่อมต่อด้วยวงจรอิเล็กทรอนิกส์จัมเปอร์ป้องกันที่ใช้สำหรับ PCB มี ฉนวน เรซิ่น PBT ทนความร้อน ฉนวนไฟฟ้า

2.12 Bluetooth Module HC-05



รูปที่ 2.17 HC-05

Bluetooth Module ที่ใช้ในการส่งข้อมูลไร้สายด้วย 2.4 GHz Bluetooth 2.0+EDR (Enhanced Data Rate) มีขนาดเล็กและใช้พลังงานต่ำ สามารถต่ออินเตอร์เฟสกับคอมพิวเตอร์ด้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UART Interface เพื่อใช้งาน AT Command
คุณสมบัติ Bluetooth Module
ความถี่ 2.4 GHz Bluetooth 2.0+EDR
สามารถตั้งค่าเป็นได้ทั้ง Master และ Slave
แรงดัน 3.3 V. (1.8-3.6 V)
รองรับ Baud rate ที่ 9600,19200,38400,57600,115200,230400,460800
Built-in antenna
ขนาด 27.0*12.7*2.2 mm.
UART Interface สามารถปรับค่า Baud rate ได้
แก้ไขค่าต่างๆได้ด้วยคำสั่ง AT Command

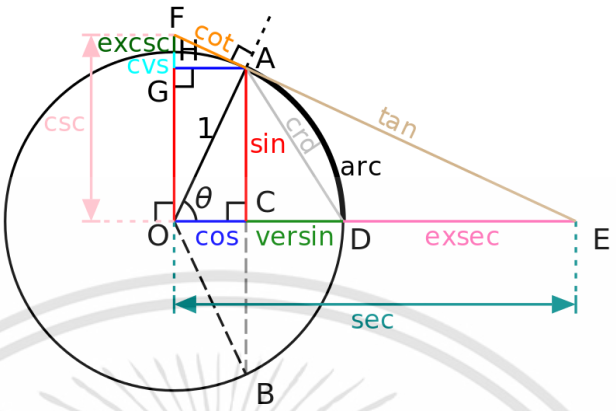
2.13 HMC5883L

รูปที่ 2.18 HMC5883L GY-271

3-Axis Digital Compass IC (HMC5883L) GY-271 Sensor GY-271 Arduino Sensor ไจโร-ความเร่ง สำหรับวัดการเคลื่อนไหว แนวตั้งแนว นอน การเอียงแบบ 3 แกน (3-axis) เหมาะสำหรับการนำไปประยุคต์ใช้งานการสร้างอุปกรณ์ วัดความเอียงต่างๆ เชื่อการทำเครื่องบิน โดรน (เพื่อให้การบินเสถียร) เป็นต้น การเชื่อกับ Arduino หรือ ไมโครคอนโทรลเลอร์ชนิดๆ อื่นๆ PIC MCS-51 STM โดนผ่านการเชื่อมต่อแบบ digital I2C (I2C Interface)
พิกัดของเป้าหมายและทิศทางของรถที่สัมพันธ์กับทิศเหนือแม่เหล็ก ระยะทางและทิศทาง มีข้อผิดพลาดที่จะพยายามลดให้เหลือน้อยที่สุดอัลกอริธึมการควบคุม ความคลาดเคลื่อนของระยะทางสามารถลดลงได้โดยการทำให้รถมีความเร็วเชิงเส้น และความคลาดเคลื่อนของมุมโดยการให้ความเร็วเชิงมุมที่สอดคล้องกัน "สูตร Haversine" สามารถคำนวณความผิดพลาดของระยะทาง "d" และ "สูตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Forward Azimuth" สามารถคำนวณมุมราบ (waypoint_angle) ซึ่งสามารถคำนวณข้อผิดพลาดของส่วนหัว "α" ได้



รูปที่ 2.19 haversine

สูตร Haversine เป็นสูตรทั่วไปในตรีโกณมิติทรงกลมที่ใช้ในการคำนวณระยะห่างระหว่างจุดสองจุดบนพื้นผิวโลกหรือทรงกลม การคำนวณระยะห่างระหว่างจุดสองจุดบนพื้นผิวโลก ต้องมีพิกัดละติจูดและลองจิจูดของจุดสองจุด ให้สูตร Haversine ทั่วไปและเอกลักษณ์ตรีโกณมิติ Haversine ตามลำดับ เป็นสมการที่ใช้ในโค้ด

$$\mathrm{hav}\left(\frac{d}{r}\right)=\mathrm{hav}(\varphi_2-\varphi_1)+\cos(\varphi_1)\times\cos(\varphi_2)\times\mathrm{hav}(\lambda_2-\lambda_1)$$

$$\mathrm{hav}(\theta)=\sin^2\left(\frac{\theta}{2}\right)=\frac{1-\cos(\theta)}{2}$$

$$a=\sin^2\left(\frac{\Delta\varphi}{2}\right)+\cos(\varphi_1)\times\cos(\varphi_2)\times\sin^2\left(\frac{\Delta\lambda}{2}\right)$$

$$c=2\times\arctan 2(\sqrt{a},\sqrt{1-a})$$

$$d=r\times c$$

φ_1 : point 1 latitude, λ_1 : point 1 longitude

φ_2 : point 2 latitude, λ_2 : point 2 longitude

d : arc(distance), r : Earth's radius

$\Delta\varphi=\varphi_2-\varphi_1$, $\Delta\lambda=\lambda_2-\lambda_1$

รูปภาพที่ 2.20 สมการคำนวณ พิกัดละติจูดและลองจิจูด

มุมแอซิมัทถูกกำหนดให้เป็นมุม "แนวนอน" ซึ่งวัดตามเข็มนาฬิกาจากเส้นฐานเหนือหรือเส้นเมริเดียนถึงเส้นบอกทิศทาง ซึ่งกำหนดโดยจุดสองจุดเหนือพื้นผิวโลก ใช้ทิศเหนือ แอซิมัท และถ้าใช้สนามแม่เหล็กเหนือ จะได้ แอซิมัทแม่เหล็ก ที่สอดคล้องกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น Forward Azimuth หรือ Azimuth Formula สามารถคำนวณมุมระหว่างสองเส้นนี้ เส้นระหว่างตำแหน่งของรถกับทิศเหนือ และเส้นแบ่งระหว่างตำแหน่งของรถกับเป้าหมาย

$$\theta = a\tan 2(\sin\ \Delta\lambda\ \times\ \cos\ \varphi_2, \cos\varphi_1\ \times \sin\varphi_2 - \sin\varphi_1 \times \cos\varphi_2 \times \cos\Delta\lambda)$$

φ_1 : point 1 latitude, φ_2 : point 2 latitude

θ : waypoint angle or azimuth

$$\Delta\lambda = \lambda_1 - \lambda_2$$

รูปที่ 2.21 สมการการคำนวณมุมแอซิมัท

มีมุมแอซิมัทหรือมุมเวย์พอยท์ ต้องคำนวณความผิดพลาดของทิศทาง ซึ่งเป็นเพียงความแตกต่างระหว่างมุมแอซิมัทกับทิศทางของรถที่สัมพันธ์กับทิศเหนือซึ่งได้มาจากเข็มทิศดิจิทัล

heading error = waypoint angle robot car heading

เมื่อได้ค่า heading error จะทำการนำค่านี้มาเพื่อคำนวณค่าความเร็วล้อของแต่ละข้าง เพื่อเข้าสู่จุดหมาย

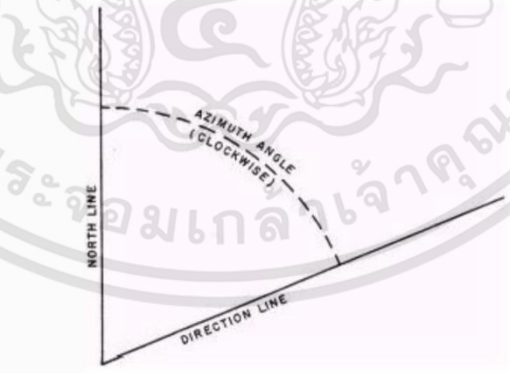


Figure 5-29.-Azimuth angle.

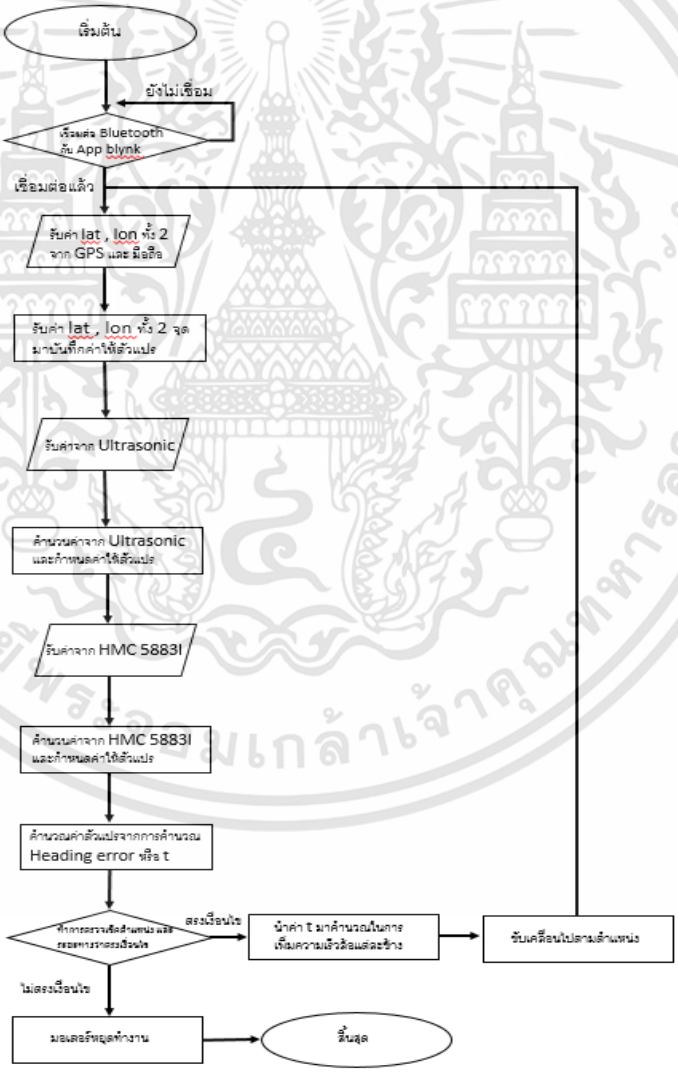
รูปภาพที่ 2.22 Azimuth angle

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและวิธีการดำเนินการ

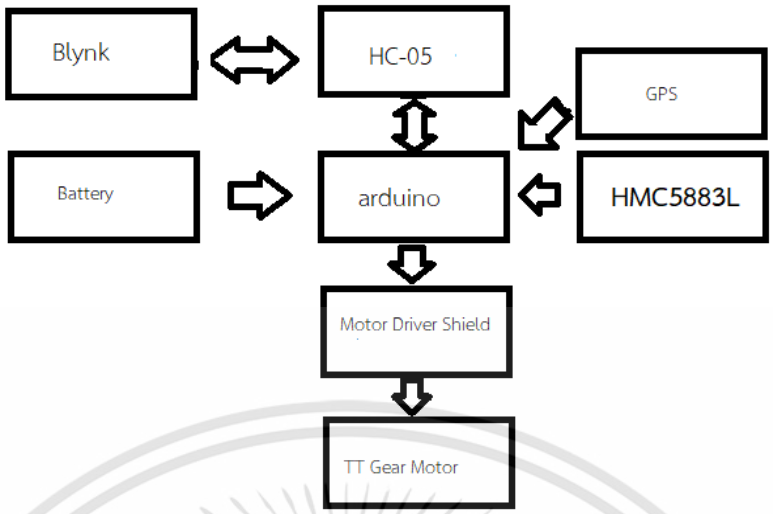
3.1 การออกแบบ

หลักการในการออกแบบรถติดตามอัจฉริยะ สามารถแบ่งส่วนการทำงานได้ 2 ส่วน คือ ส่วนของการทำงานของอุปกรณ์ ซึ่งเป็นอุปกรณ์ที่ใช้ในการติดตั้งต่าง ๆ และส่วนแอพลิเคชั่น บนหน้าจอสมาร์ทโฟน เพื่อทำการเชื่อมต่อกับสมาร์ทโฟนและแสดงผลไร้สายบนโปรแกรม แอพพลิเคชั่น Blynk การทำงานตามขั้นตอน ดังรูปที่ 3.1



รูปที่ 3.1 Flow Chart Program

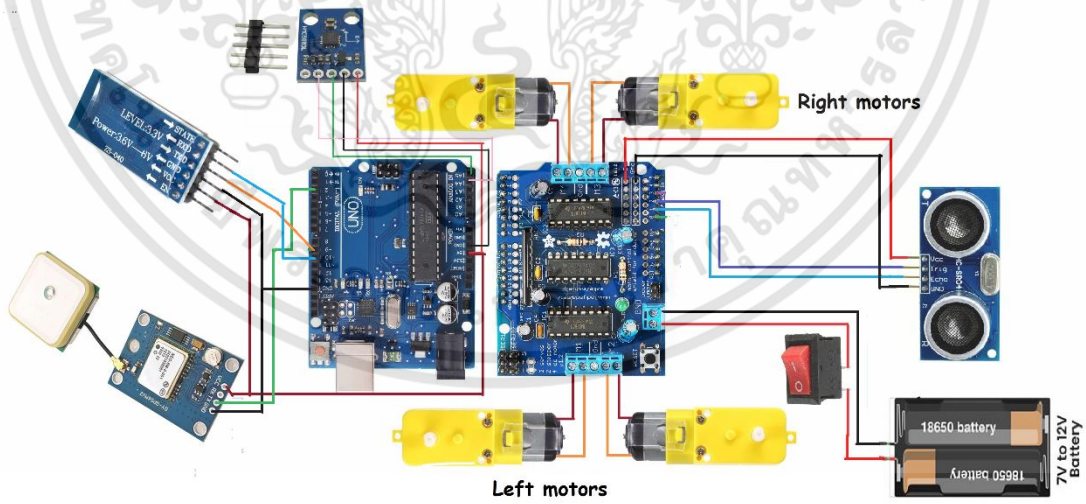
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 โครงสร้างการออกแบบและหลักการทำงาน

3.2 ขั้นตอนการทำงานในส่วนของอุปกรณ์

โดยในส่วนของอุปกรณ์ ออกแบบเพื่อทำการโครงร่าง CarTracker และสัญญาณ Bluetooth เชื่อมกันสมาร์ทโฟนไปยังส่วนของ Server Blynk เพื่อรอการนำแสดงผลต่อไป



รูปที่ 3.3 การเชื่อมต่ออุปกรณ์และการประกอบโครงรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 Arduino Uno

เป็นหัวใจหลักของระบบการสื่อสารในการเชื่อมต่อระหว่างเซนเซอร์ โดยจะทำหน้าที่ในการอ่านค่าต่าง ๆ จากเซนเซอร์แล้วส่งข้อมูลไปยัง Server Blynk ซึ่งใช้การสื่อสารแบบ MQTT ติดต่อระหว่าง Server กับ Microcontroller ภายใน Arduino Uno

3.2.2 Motor Driver Shield

ในส่วนอุปกรณ์อิเล็กทรอนิกส์มีขนาดเท่ากับบอร์ด Arduino ต่อเชื่อมกับบอร์ด Arduino Uno สามารถใช้ขับ DC มอเตอร์ขนาดเล็ก 4 ตัว โดยกระแสต่อเนื่องได้ไม่เกิน 600 mA รับแรงดันที่จ่ายให้กับมอเตอร์ 4.5 ถึง 36 V ให้มอเตอร์ทำงานอย่างมีประสิทธิภาพ

3.2.3 TT Gear Motor

มอเตอร์เกียร์ DC ขนาดเล็ก รับแรงดันไฟฟ้า 1.5V ถึง 24V ต่อขั้วบวก ขั้วลบ เข้ากับ Motor Driver Shield ทำการส่งแรงบิดโหลดของมอเตอร์ไปขับเคลื่อนล้อให้ทำงานตามดังโปรแกรม

3.2.4 Ultrasonic Sensor

ทำหน้าที่ตรวจจับวัตถุหรือสิ่งกีดขวาง โดยจะปล่อยคลื่น Ultrasonic ให้กระทบกับวัตถุ จากนั้นรอคลื่น Ultrasonic สะท้อนกับมาที่เซ็นเซอร์เพื่อคำนวณหาระยะทางที่วัด ส่งไปยังบอร์ด arduino Uno เพื่อคำนวณหาค่าระยะที่ได้ และสั่งให้มอเตอร์เกียร์ DC หยุดทำงานตามคำสั่ง

3.2.5 Bluetooth Module

Module Bluetooth HC-05 ใช้ในการเชื่อมต่อกับสมาร์ทดีไวซ์ ให้สามารถดีไวซ์สามารถสื่อสารกับไมโครคอนโทรเลอร์ ผ่าน Serial port ให้ใช้โหมด Master และโหมด Slave การตั้งค่าต่างๆ ทำได้ผ่าน AT Command ใช้แหล่งจ่ายไฟที่ 5V ขา Tx ต่อเข้าขา 9 Rx ต่อเข้าขา 10 และส่งสัญญาณไปยังอุปกรณ์ที่เชื่อม เพื่อรับคำสั่งให้ทำงาน

3.2.6 GPS Module

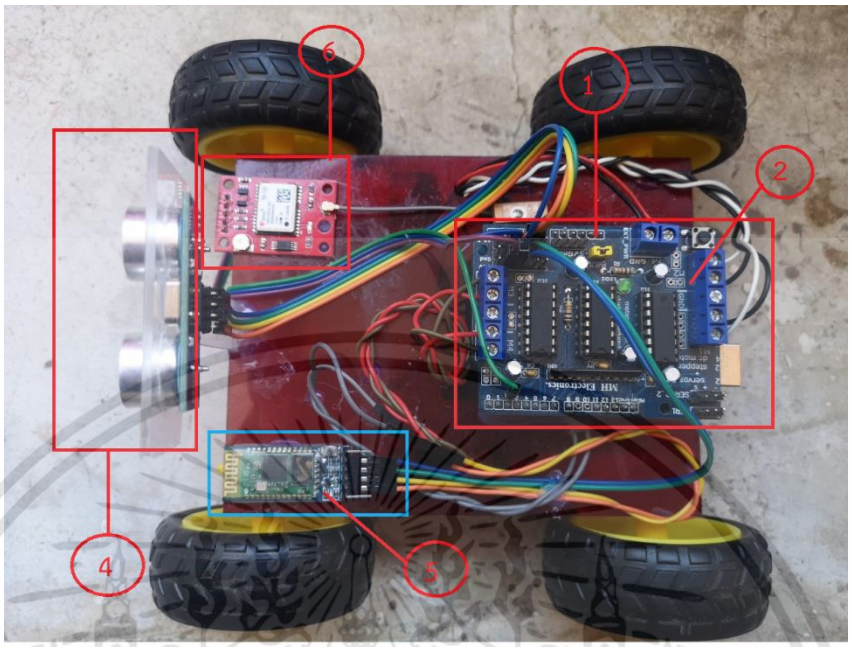
GPS Module ทำหน้าที่บอกตำแหน่งรถหรือนำทางในการเคลื่อนที่ของรถไปในที่กำหนด การต่อขาTx เข้าขา 2 ของบอร์ด Arduino Uno โดยคำสั่งจากบอร์ด

3.2.7 Magnetometer Sensor Module

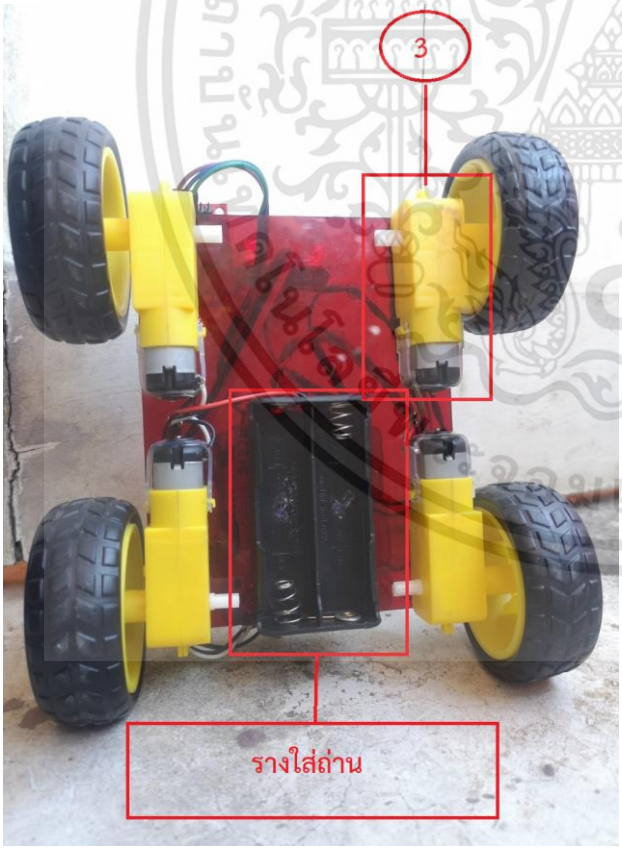
ทำหน้าที่วัดการเคลื่อนไหว แนวตั้ง แนวนอน การเอียงแบบ 3 แกน และมันสามารถที่จะวัดสนามแม่เหล็ก ซึ่งทำให้มันกลายเป็นเข็มทิศและนำค่าที่อ่านได้มาคำนวณความเร็วของล้อในรถ

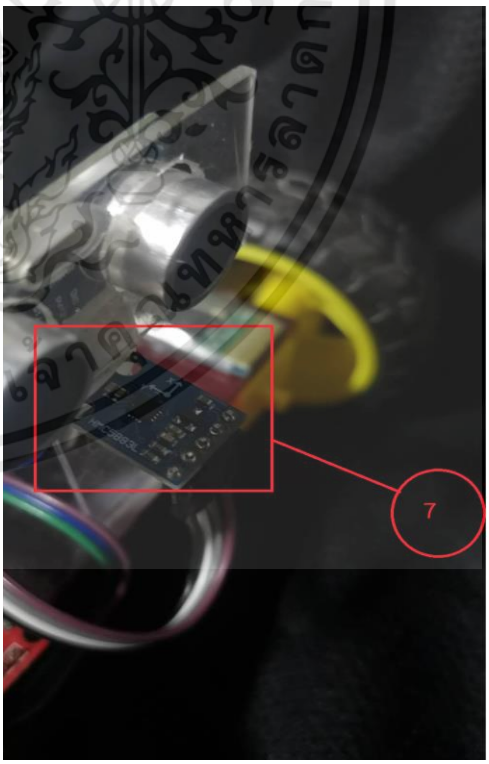
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.8 ตำแหน่งการวางอุปกรณ์



รูปที่ 3.4 ตำแหน่งการติดตั้งอุปกรณ์ภายในตัวรถ





รูปที่ 3.5 ตำแหน่งการติดตั้งอุปกรณ์ใต้ตัวรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบ CarTracker เป็นการออกแบบอุปกรณ์รถโมเดลที่ทำให้ CarTracker มีประสิทธิภาพมากขึ้นโดยจะมีแอพพลิเคชันควบคุมการทำงานของตัวรถ เพียงแต่เป็นรถรุ่นโมเดลหรือทดลองการใช้งานควบคู่กับแอพพลิเคชัน โดยมีตำแหน่งการจัดวางอุปกรณ์ดังนี้

หมายเลข 1 คือ Arduino Uno

หมายเลข 2 คือ Motor Driver Shield

หมายเลข 3 คือ TT Gear Motor

หมายเลข 4 คือ Ultrasonic Sensor

หมายเลข 5 คือ Bluetooth Module

หมายเลข 6 คือ GPS Module

หมายเลข 7 คือ Magnetometer Sensor Module

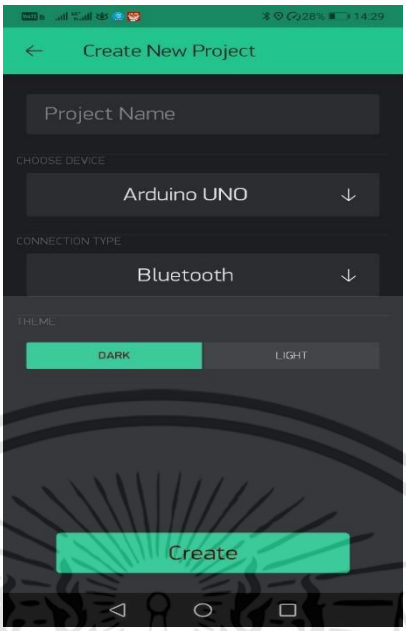
3.3 ขั้นตอนการทำงานในส่วนของการแสดงผล

ในส่วนของการแสดงผลจะออกแบบเพื่ออ่านค่าสถานะต่าง ๆ คำสั่งของโปรแกรมเพื่อทำการคำนวณ และแสดงผลไร้สายบนแอพพลิเคชั่น Blynk

3.3.1 แอพพลิเคชั่น Blynk

ในการใช้งาน blynk สามารถใช้งานได้ทั้ง IOS/Android เมื่อทำการติดตั้งโปรแกรมลงบนโทรศัพท์แล้วให้ทำการเข้าแอพพลิเคชั่นและลงชื่อเข้าใช้ตามปกติ หลังจากนั้นให้ทำการสร้างโปรแกรมโดยกดเข้าไปที่ New Project เพื่อจะได้ใช้งานในขั้นต่อไปเมื่อเข้ามาที่หน้าถัดไปนั้นจะเป็นการใส่ชื่อ Project ของเรา ในช่องต่อมาจะเป็นการเลือกตัว Controller ว่าเราใช้งานประเภทไหน และช่องต่อมาจะเป็นการเลือกวิธีการติดต่อใช้งานกับแอพพลิเคชั่นว่าเราจะเลือกวิธีการติดต่อแบบไหน โดยการเลือกครั้งนี้จะเป็น controller Arduion UNO และเลือกการติดต่อแบบ Bluetooth ในการสื่อสารนั้นเองและจากนั้นก็ทำการ Create

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



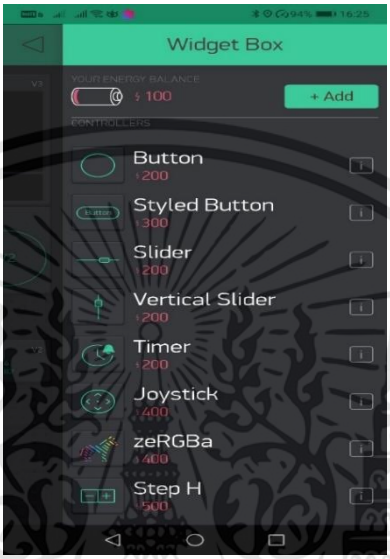
รูปภาพที่ 3.6 หน้าต่างโปรแกรม Blynk

เมื่อทำการสร้างแล้วจะขึ้นหน้าต่างที่จะบอกกับเราว่าจะส่ง Token ไปให้เราทาง E – mail ให้นำ Token ที่เราได้จากที่แอพพลิเคชั่นส่งมาให้นั้นมาใส่ลงในโปรแกรมเพื่อให้เราสามารถติดต่อกับ blynk สร้างไว้ได้หลังจกานั้นก็จะเป็นหน้าโปรแกรมเปล่าให้พร้อมใช้งานตามการทำงานของเรา

รูปภาพที่ 3.7 แสดงหน้าจอการใช้งานของแอพพลิเคชั่น Blynk

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราปัดโทรศัพท์จากขาวไปทางด้านซ้ายจะมีอุปกรณ์ที่เราสารามใช้งานได้ตามที่เราต้องการและทำการกดเพื่อนำมาวางไว้ตำแหน่งที่เราต้องการเพื่อ โดยแต่ละปุ่มนั้นสามารถที่จะตั้งชื่อและเลือกขาที่เราจะใช้งานได้ไม่ว่าจะเป็นขา digital หรือขา Analog และมีขา virtual ที่เป็นของตัว blynk เองซึ่งสามารถนำมาเขียนโปรแกรมเพื่อมาใช้งานในแอพได้เหมือนกัน



รูปภาพที่ 3.8 หน้าต่างบล็อคหรือปุ่มที่ต้องการเลือก

เมื่อเราทำการจัดการใส่อุปกรณ์ต่าง ๆ ที่เราต้องการแล้ว เมื่อถึงเวลาที่ต้องการติดต่อกับตัวแอพให้ทำการใส่ Token ที่ได้จาก E – mail มาใส่ในช่อง “YourAuthToken” เพื่อเป็นการเช็คความถูกต้องว่าทำการติดต่อถูกต้องหรือป่าว เมื่อใส่ลงในช่องแล้วทำการ upload ลงใน Controller และเชื่อต่อกับแอพได้เลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial_HC05_HC06 | Arduino 1.8.16 (Windows Store 1.8.51.0)

File Edit Sketch Tools Help

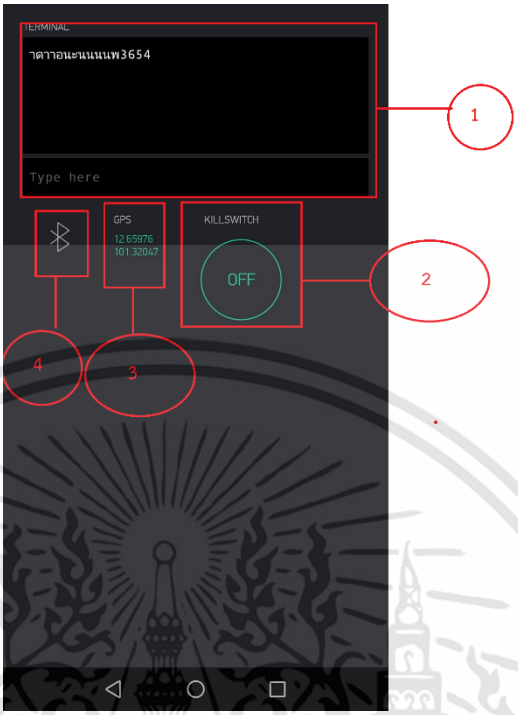
Serial_HC05_HC06

```
#define BLYNK_USE_DIRECT_CONNECT

// You could use a spare Hardware Serial on boards that have it (like Mega)
#include <SoftwareSerial.h>
SoftwareSerial DebugSerial(2, 3); // RX, TX

#define BLYNK_PRINT DebugSerial

/* Fill-in your Template ID (only if using Blynk.Cloud) */
//#define BLYNK_TEMPLATE_ID   "YourTemplateID"


#include <BlynkSimpleSerialBLE.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "YourAuthToken";

void setup()
{
  // Debug console
  DebugSerial.begin(9600);

  DebugSerial.println("Waiting for connections...");

  // Blynk will work through Serial
  // 9600 is for HC-06. For HC-05 default speed is 38400
  // Do not read or write this serial manually in your sketch
  Serial.begin(9600);
  Blynk.begin(Serial, auth);
}
```

รูปภาพที่ 3.9 โค้ดโปรแกรม Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ตำแหน่งปุ่มกด Blynk



รูปภาพที่ 3.10 โปรแกรมที่พร้อมใช้งาน

หมายเลข 1 คือ หน้าจอแสดงผล

หมายเลข 2 คือ ปุ่มหยุดการทำงาน

หมายเลข 3 คือ ปุ่มรีสตาร์ทค่า GPS

หมายเลข 4 คือ Bluetooth

3.3.3 การเขียนโปรแกรม

ในการใช้งานและให้สามารถรับค่าหรือรันคำสั่งได้นั้น จำเป็นที่ต้องมีตัวโปรแกรมที่จะสามารถใช้ในการกำหนดเงื่อนไขและรูปแบบในการทำงานของตัวรถของเราให้ทำงานตาม รูปที่ 3.1 ได้นั้นต้องเขียนโปรแกรมเข้าไปใน Controller ซึ่งในการเขียนโปรแกรมครั้งนี้ใช้โปรแกรม arduino ide ในการเขียนและทำให้รถสามารถวิ่งตามโปรแกรมที่ออกแบบได้ตามภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

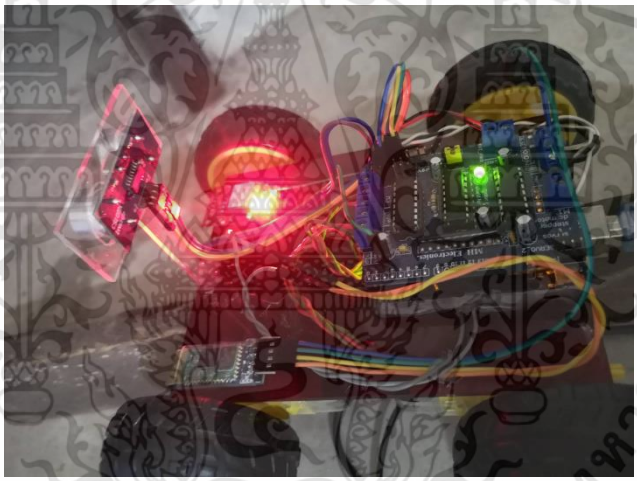
บทที่ 4
ผลการทดลอง

4.1 กล่าวนำ

การทดลองระบบรถติดตามในสิ่งประดิษฐ์ที่สร้างมา และแสดงผลบนหน้าจอโทรศัพท์ ประกอบไปด้วยการเคลื่อนที่ของตัวรถติดตาม รวมทั้งการควบคุมจากโทรศัพท์และแสดงผลผ่านแอพพลิเคชัน Blynk

4.2 ขั้นตอนการทดลอง

4.2.1.เปิดสวิตช์รอสัญญาณ GPS ให้ GPS ติดต่อกับดาวเทียม โดยดูสถานะไฟ Module GPS ให้อยู่สถานะพร้อมใช้งาน ถ้าติดต่อได้แล้วจะกระพริบ ถ้าติดต่อไม่ได้ไฟจะติดค้าง



รูปภาพที่ 4.1 รถสถานะพร้อมใช้งาน

4.2.2.เมื่อ GPS ติดต่อสื่อสารกับดาวเทียมได้แล้ว เข้าแอพลิเคชัน Blynk ทำการเชื่อม Bluetooth จากตัวโมดูล HC-05 ในการเชื่อมต่อให้ดูสัญญาณไฟโมดูล ถ้าโมดูลมีไฟกระพริบถี่แสดงว่ายังไม่ได้เชื่อมต่อ Bluetooth แล้วถ้ามีไฟกระพริบช้าแสดงว่าเชื่อมต่อ Bluetooth สำเร็จแล้ว

4.2.3 หลังจากที่ทำการเชื่อม Bluetooth เรียบร้อยสามารถ Monitor ดูที่โปรแกรมได้ว่าเชื่อมต่อเรียบร้อยดีไหม และยังสามารถเขียนที่ Terminal ของตัว blynk สามารถติดต่อกับมา Controller ได้หรือไม่ เมื่อลองตอบโต้ดูแล้วสามารถเชื่อมต่อได้ และเมื่อเราอยู่ในระยะของเงื่อนไข GPS ถ้าน้อยกว่า 6 m รถจะไม่ทำงาน แต่ถ้ามากกว่า 6 m รถจะไปตามตำแหน่งที่อยู่ของมือถือ และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้ามีสิ่งกีดขวางอยู่ข้างหน้าและอยู่ในระยะ 50 cm รถก็จะหยุดแต่ทุกอย่างจะมีความหน่วง 5 s ในการคำนวณ เพราะการส่งค่าตำแหน่งของมือถือจะมีความหน่วง

4.3 ผลการทดลอง

ขั้นตอนแรก ผลไม่เป็นตามที่เราต้องการ รันโปแกรมเข้าไปแล้ว ทำให้รถวิ่งแต่ยังไม่ติดคนอย่างที่ต้องการ GPS ไม่สามารถเชื่อมต่อ GPS ให้ ทำให้ประสิทธิไม่เสถียร

ขั้นตอนสอง รถมีการเคลื่อนที่ได้ GPS ใช้งานได้ปกติ แต่ยังนำทางให้เคลื่อนที่รถยังไม่ตรงตามเป้าหมาย เซ็นเซอร์ตรวจจับคนใช้ได้ปกติในระยะที่ตั้งค่าไว้

ขั้นตอนสาม รถมีการเคลื่อนตามที่เราต้องการ GPS ใช้งานได้ปกติ แต่จะมีความไม่เสถียร เซ็นเซอร์ทำการตรวจจับได้ในระยะที่ตั้งค่าไว้ ทำให้รถหยุด แล้วรันคำสั่งต่อไปก็ทำงานได้ตามปกติ

เนื่องจากรถเป็น 4 ล้อ และไม่ได้ใช้วิธีการหมุนล้อหน้าแบบรถยนต์ จึงใช้วิธีการลดความเร็วหรือเพิ่มความเร็วของล้อแต่ละข้างแทนเพื่อใช้ในการเลี้ยวและยังเป็นการวิ่ง 4 ล้อเหมือนเดิม ตัวรถมีไว้เพื่อบอกตำแหน่งและค่าพิกัดของรถ เพื่อใช้ในการคำนวนระยะทางกับตำแหน่งของคน ในส่วนระยะตรวจจับ ได้มีการเขียนไว้ในโปรแกรมประมาณ 6 m เมื่อทำการทดลองก็จะให้คนห่างกับรถโดยประมาณ 6 m แต่เนื่องการส่งค่าที่มีการหน่วงของเวลาจากแอพพลิเคชันที่ 5 s ทำให้การคำนวณล่าช้าและความห่างจึงน้อยกว่า 6 m ที่ตั้งไว้ในโปรแกรม ในส่วนการจับของ GPS นั้นระยะที่จับไม่ได้จะเป็นที่อับสัญญาณ ซึ่งทำให้ GPS นั้นติดต่อกับดาวเทียมและหาตำแหน่งเกิดความไม่เสถียร ตัวคนต้องถือ GPS อีกตัวนึงเพื่อให้บอกตำแหน่งและไปคำนวณค่าตำแหน่ง แต่ตัว GPS ที่ถือนั้นก็คือมือถือ เราใช้ตำแหน่งจากมือถือไปบอกกับตัวรถ

HMC 5883l ในการทดลองใช้งานครั้งแรกที่ใช้งานนั้นสามารถใช้งานได้ตามปกติ แต่ในการติดตั้งกับตัวรถจะมีผลกับการทดลองใช้งานตัวเข็มทิศ เมื่อทำการติดกับตัวรถเลยโดยที่ไม่มีพื้นที่ว่างเผื่อไว้ให้จะทำให้การอ่านค่านั้นผิดเพี้ยนไม่สามารถอ่านค่าได้อย่างการทดลองแรก หลังจากทำการติดตั้งโดยเผื่อไว้แล้วจะสามารถใช้งานได้ตามที่ทดลองก่อนติดตั้ง โดยตัว HMC 5883l ในการใช้งานเราจะนำค่าที่ได้มาคำนวณและเข้ากับสมการ Azimuth Formula เพื่อใช้ในการคำนวณของล้อแต่ข้าง

4.4 อุปสรรคในการทดลอง

4.3.1 โปรแกรม Error ในการทดลอง

4.3.2 GPS ที่ติดต่อกับดาวเทียม มีความไม่เสถียรภาพ ในการทดลองรถ CarTracker ทำให้เกิดการหาสัญญาณไม่สมบูรณ์

4.3.3 เนื่องจากการส่งค่าเรียลไทร์ ของGPS ในมือถือ มีความหน่วง 5 m ทำให้โปรแกรมมีการทำงานล่าช้าและคำนวณคาดเคลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลอง

จากการทดลองเป็นการทดสอบรถติดตามตัวบุคคล จากจุดหนึ่งไปยังจุดหนึ่ง โดยใช้ GPS Model NEO6M ในการนำเส้นทางให้กับตัวรถติดตาม และมีเซนเซอร์ที่สามารถตรวจจับการเครื่องที่ของสิ่งรอบๆตัวรถติดตาม เพื่อไม่ให้เกิดความเสียหายและสร้างความแม่นยำให้กับรถติดตาม สามารถติดตามการทำงานของรถติดตามตัวบุคคลได้ผ่านโทรศัพท์ เป็นความสะดวกสบายในการใช้งาน ตัวรถติดตามจะออกห่างตามที่โปรแกรมได้ตั้งไว้ ตามที่เราต้องการ

ในขั้นตอนแรกจำเป็นที่จะต้องให้ Bluetooth HC05 เชื่อมต่อกับโทรศัพท์ส่วนบุคคลเพื่อเปิดการทำงานของตัวรถติดตาม เมื่อทำการเชื่อมต่อเรียบร้อยแล้ว จากนั้นให้ทำการเขียนโปรแกรมเขียนคำสั่งทั้งหมดที่จำเป็นในการควบคุมตัวรถติดตามของเรา จากนั้นทำการโหลดโปรแกรมเข้าในตัวรถติดตาม ต่อไปคือทำการทดสอบใช้ตัวรถติดตามว่าทำงานได้ปกติหรือไม่ ตรวจสอบและสังเกตการทำงานของตัวรถติดตาม เพื่อที่จะนำการทำงานต่างๆไปวิเคราะห์และพัฒนาโครงการนี้ต่อไป

ในส่วนของแอพพลิเคชัน Blynk จะแสดงพารามิเตอร์ต่างๆที่ต้องการคำสั่งทั้งหมด มีหน้าจอแสดงผลการทำงานของตัวรถติดตาม เพื่อความสะดวกสบายของผู้ใช้งาน ทดสอบโดยการใช้แอพพลิเคชันควบคุมตัวรถติดตามว่าสามารถติดตามตัวผู้ใช้งานในระยะที่กำหนดได้หรือไม่ และวิเคราะห์การทำงานกับประสิทธิภาพของตัวรถและแอพพลิเคชันในการพัฒนาครั้งต่อไป

ในการทดลองนี้มีจุดประสงค์หลักคือ การนำความสามารถต่างๆที่มีอยู่อุปกรณ์อิเล็กทรอนิกส์ เช่น เซนเซอร์ หรือจีพีเอส แถมยังมีการใช้งานแอพพลิเคชันที่จะสามารถเรียกใช้งานรถติดตามเมื่อไรก็ได้ ในครั้งนี้จะใช้โปรแกรม Arduino IDE Program และแอพพลิเคชัน Blynk มาใช้งานร่วมกัน เพื่อที่จะได้ผลลัพธ์ตามที่ต้องการสร้างความสะดวกสบายให้แก่ผู้ใช้งาน โครงการนี้ได้นำความรู้ในศาสตร์ด้านต่างๆที่ได้เรียนรู้มาใช้ในสาขาวิชาวิศวกรรมการวัดคุม หลักสูตรบัณฑิตพันธุ์ใหม่ที่เน้นทางด้านอุปกรณ์อิเล็กทรอนิกส์และการเขียนโปรแกรม เช่น รายวิชาไมโครคอนโทรลเลอร์ และอีกมากมายมาประยุกต์ใช้ในการทำโครงงานนี้ให้สำเร็จลุล่วงไปได้ด้วยดี

โครงการนี้สามารถทำรถให้ติดตามตัวบุคคลได้ โดยใช้ GPS ในการนำทางให้แก่ตัวรถติดตาม และอนาคตสามารถนำโปรเจคไปต่อยอดสร้างสิ่งประดิษฐ์ติดตาม เช่น กระเป๋าเดินทางติดตามตัวคน เครื่องจักรในโรงงานที่สามารถติดตามคนได้ หรืออุปกรณ์เครื่องมือที่ใช้แบกกระเป๋าเดินทางอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อลำเลียงขึ้นเครื่องบินโดยที่ไม่ต้องใช้คนเข็นให้วุ่นวายอีกต่อไป ทั้งหมดนี้คือสิ่งประดิษฐ์ที่ช่วยอำนวยความสะดวกสบายให้แก่ผู้ใช้งานในงานที่จำเป็นต้องใช้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] https://create.arduino.cc/projecthub/mohammadsohail0008/human-following-bot-070eaa ค้นวันที่ 16 ธันวาคม 2564.

[2] https://techatronic.com/human-following-robot-using-arduino/. ค้นวันที่ 16 ธันวาคม 2564

[3] https://www.instructables.com/How-to-Make-a-Human-Following-Robot-With-Arduino/ ค้นวันที่ 16 ธันวาคม 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมหลัก

```
#define BLYNK_PRINT Serial

#include <Wire.h>
#include <Adafruit_HMC5883_U.h>
#include <AFMotor.h>
#include <Adafruit_Sensor.h>
#include <Servo.h>
#include <SoftwareSerial.h>
#include <BlynkSimpleSerialBLE.h>
#include "./TinyGPS.h"
#include "./Definitions.h"

AF_DCMotor motor1(1, MOTOR12_64KHZ);
AF_DCMotor motor2(2, MOTOR12_64KHZ);
AF_DCMotor motor3(3, MOTOR34_64KHZ);
AF_DCMotor motor4(4, MOTOR34_64KHZ);


TinyGPS gps;


bool enabled = false;

SoftwareSerial bluetoothSerial(BLUETOOTH_TX_PIN, BLUETOOTH_RX_PIN);
SoftwareSerial nss(GPS_TX_PIN, 255);

Adafruit_HMC5883_Unified mag = Adafruit_HMC5883_Unified(12345);

GeoLoc checkGPS() {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Serial.println("Reading onboard GPS: ");
  bool newdata = false;
  unsigned long start = millis();
  while (millis() - start < GPS_UPDATE_INTERVAL) {
    if (feedgps())
      newdata = true;
  }
  if (newdata) {
    return gpsdump(gps);
  }

  GeoLoc coolerLoc;
  coolerLoc.lat = 0.0;
  coolerLoc.lon = 0.0;

  return coolerLoc;
}


GeoLoc gpsdump(TinyGPS &gps) {
  float flat, flon;
  unsigned long age;

  gps.f_get_position(&flat, &flon, &age);

  Blat = flat;
  Blon = flon;

  GeoLoc coolerLoc;
  coolerLoc.lat = flat;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  coolerLoc.lon = flon;

  Serial.print(coolerLoc.lat, 7); Serial.print(", "); Serial.println(coolerLoc.lon, 7);
  return coolerLoc;
}

bool feedgps() {
  while (nss.available()) {
    if (gps.encode(nss.read()))
      return true;
  }
  return false;
}
BLYNK_WRITE(V1) {
  enabled = !enabled;

  stop();
}

BLYNK_WRITE(V2) {
  GpsParam gps(param);

  Serial.println("Received remote GPS: ");

  Serial.print(gps.getLat(), 7); Serial.print(", "); Serial.println(gps.getLon(), 7);


  slat = gps.getLat();
  slon = gps.getLon();
  GeoLoc phoneLoc;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  phoneLoc.lat = gps.getLat();
  phoneLoc.lon = gps.getLon();
  geoBearing(); geoHeading();
   t = geoBear - geoHead ;
  LocationR();
  Serial.print("distanceHM    ");
  Serial.println(distance,7);

  Serial.print("t :    ");
  Serial.println(t,7);

  driveTo(phoneLoc, GPS_STREAM_TIMEOUT);

    if (Blat != 0 && Blon != 0 && slat != 0 && slon != 0 && distance > 6 && enabled) {
    //Start move loop here
      drive();
      Serial.print("drive :    ");
    }
    if (distance < 6 || distanceUT < 50 ){
     Serial.print(" stop    ");
    stop();
    }
}
BLYNK_WRITE(V3) {
  Serial.print("Received Text: ");
  Serial.println(param.asStr());

  String rawInput(param.asStr());
  int colonIndex;
  int commaIndex;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  do {
    commaIndex = rawInput.indexOf(',');
    colonIndex = rawInput.indexOf(':');

    if (commaIndex != -1) {
      String latStr = rawInput.substring(0, commaIndex);
      String lonStr = rawInput.substring(commaIndex+1);

      if (colonIndex != -1) {
        lonStr = rawInput.substring(commaIndex+1, colonIndex);
      }

      float lat = latStr.toFloat();
      float lon = lonStr.toFloat();

      if (lat != 0 && lon != 0) {
        GeoLoc waypoint;
        waypoint.lat = lat;
        waypoint.lon = lon;

      }
    }

    rawInput = rawInput.substring(colonIndex + 1);

  } while (colonIndex != -1);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void stop() {
  motor1.run(RELEASE);
  motor2.run(RELEASE);
  motor3.run(RELEASE);
  motor4.run(RELEASE);

}

void drive() {
  int fullSpeed = 230;
  int stopSpeed = 0;

  // drive to location
  int s = fullSpeed;
  if ( distance < 8 ) {
    int wouldBeSpeed = s - stopSpeed;
    wouldBeSpeed *= distance / 8.0f;
    s = stopSpeed + wouldBeSpeed;
  }

  int autoThrottle = constrain(s, stopSpeed, fullSpeed);

    float T = t;
  while (T < -180) T += 360;
  while (T >  180) T -= 360;
  distance = gps.distance_between (slat,slon,Blat,Blon);
  Serial.print("turn: ");
  Serial.println(T);
  Serial.print("original: ");
  Serial.println(t);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  float t_modifier = (180.0 - abs(T)) / 180.0;
  float autoSteerA = 1;
  float autoSteerB = 1;

  if (T < 0) {
    autoSteerB = t_modifier;
  } else if (T > 0){
    autoSteerA = t_modifier;
  }

  Serial.print("steerA: "); Serial.println(autoSteerA);
  Serial.print("steerB: "); Serial.println(autoSteerB);
   Serial.print("autoThrottle    ");
  Serial.println(autoThrottle,7);

  int speedA = (int) (((float) autoThrottle) * autoSteerA);
  int speedB = (int) (((float) autoThrottle) * autoSteerB);

  setSpeedMotorA(speedA);
  setSpeedMotorB(speedB);
}

void setSpeedMotorA(int speed) {
  motor1.run(FORWARD);
  motor2.run(BACKWARD);
  // set speed to 200 out of possible range 0~255
   Serial.print("speedA     ");
  Serial.println(speed,7);
   Serial.print("speed1     ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Serial.println(speed + MOTOR_A_OFFSET,7);
  motor1.setSpeed(speed + MOTOR_A_OFFSET);
  motor2.setSpeed(speed + MOTOR_A_OFFSET);
}

void setSpeedMotorB(int speed) {
  motor3.run(BACKWARD);
  motor4.run(FORWARD);

  // set speed to 200 out of possible range 0~255
  Serial.print("speedB     ");
  Serial.println(speed,7);
   Serial.print("speed2     ");
   Serial.println(speed + MOTOR_B_OFFSET,7);
  motor3.setSpeed(speed + MOTOR_B_OFFSET);
  motor4.setSpeed(speed + MOTOR_B_OFFSET);
}

void driveTo(struct GeoLoc &loc, int timeout) {
  nss.listen();
  GeoLoc coolerLoc = checkGPS();
  bluetoothSerial.listen();
  }

void LocationR() {

   digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);

  distanceUT = duration * 0.034 / 2;

  Serial.print("Distance: ");
  Serial.print(distanceUT);
  Serial.println(" cm");
}


void displayCompassDetails(void)
{
  sensor_t sensor;
  mag.getSensor(&sensor);
  Serial.println("------------------------------------");
  Serial.print  ("Sensor:       "); Serial.println(sensor.name);
  Serial.print  ("Driver Ver:   "); Serial.println(sensor.version);
  Serial.print  ("Unique ID:    "); Serial.println(sensor.sensor_id);
  Serial.print  ("Max Value:    "); Serial.print(sensor.max_value); Serial.println(" uT");
  Serial.print  ("Min Value:    "); Serial.print(sensor.min_value); Serial.println(" uT");
  Serial.print  ("Resolution:   "); Serial.print(sensor.resolution); Serial.println(" uT");
  Serial.println("------------------------------------");
  Serial.println("");
  delay(500);
}

void setupCompass() {
   /* Initialise the compass */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if(!mag.begin())
  {
    /* There was a problem detecting the HMC5883 ... check your connections */
    Serial.println("Ooops, no HMC5883 detected ... Check your wiring!");
    while(1);
  }

  /* Display some basic information on this sensor */
  displayCompassDetails();
}

void geoHeading() {
  /* Get a new sensor event */
  sensors_event_t event;
  mag.getEvent(&event);

  // Hold the module so that Z is pointing 'up' and you can measure the heading with x&y
  // Calculate heading when the magnetometer is level, then correct for signs of axis.
  float heading = atan2(event.magnetic.y, event.magnetic.x);

  // Offset
  heading -= DECLINATION_ANGLE;

  // Correct for when signs are reversed.
  if(heading < 0)
    heading += 2*PI;

  // Check for wrap due to addition of declination.
  if(heading > 2*PI)
    heading -= 2*PI;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  // Convert radians to degrees for readability.
  float headingDegrees = heading * 180/M_PI;

  // Map to -180 - 180
  while (headingDegrees < -180) headingDegrees += 360;
  while (headingDegrees >  180) headingDegrees -= 360;
  Serial.print("headingDegrees :   ");
  Serial.println(headingDegrees,7);
  geoHead = headingDegrees;
  return headingDegrees;
}

void geoBearing() {
  float y = sin(Blon-slon) * cos(Blat);
  float x = cos(slat)*sin(Blat) - sin(slat)*cos(Blat)*cos(Blon-slat);        //double slat,slon; //
มือถือ 1    double Blat,Blon; //หุ่นยนต์ 2
  Serial.print("geoBearing :   ");
  Serial.println(atan2(y, x) * RADTODEG,7);
  geoBear = atan2(y, x) * RADTODEG ;
  return atan2(y, x) * RADTODEG;
}


void setup()
{
   // Compass
  setupCompass();

  pinMode(LED_BUILTIN, OUTPUT);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  digitalWrite(LED_BUILTIN, HIGH);

  Serial.begin(4800);

  nss.begin(9600);

  //Bluetooth
  bluetoothSerial.begin(9600);
  Blynk.begin(bluetoothSerial, auth);

  motor1.run(RELEASE);
  motor2.run(RELEASE);
  motor3.run(RELEASE);
  motor4.run(RELEASE);

   pinMode(trigPin, OUTPUT);
   pinMode(echoPin, INPUT);

}


void loop()
{  distance = gps.distance_between (slat,slon,Blat,Blon);
  Blynk.run();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมย่อย

```
// Blynk Auth
char auth[] = "8fbIOYQQH58Ilp2poYzANdOVW7jeljm_";

  double slat,slon; //มือถือ 1
  double Blat,Blon; //หุ่นยนต์ 2
  double distance;

 #define echoPin A1
 #define trigPin A2

long duration;
int distanceUT;
float t;
float geoHead;
float geoBear;
#define SERVO_PIN 13

#define GPS_TX_PIN 2

#define BLUETOOTH_TX_PIN 9
#define BLUETOOTH_RX_PIN 10

#define MOTOR_A_OFFSET 40
#define MOTOR_B_OFFSET 0

#define DECLINATION_ANGLE -0.0127

#define GPS_UPDATE_INTERVAL 1000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define GPS_STREAM_TIMEOUT 18

#define GPS_WAYPOINT_TIMEOUT 45

#define RADTODEG 57.2957795130823208767
```

```
struct GeoLoc {
  float lat;
  float lon;
};
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้